

---

# Tabla de contenido

Introduction	1.1
Caso de aplicación de Linked Data	1.2
De Data a Linked Data	1.3
Ejemplo de microformato	1.4
Ejemplos de vocabularios útiles	1.5
Ejemplo de conversor de sintaxis RDF	1.6
Ejemplo de parseo programático de RDF	1.7
Ejemplo de inferencia de relaciones	1.8
Ejemplo de extracción de RDFa de una página	1.9
Ejemplos de consultas y exploración en DBPedia	1.10

# Introducción

En este documento podrás encontrar ejemplos prácticos relacionados con Linked Data.

# Caso de aplicación de Linked Data: The Open University

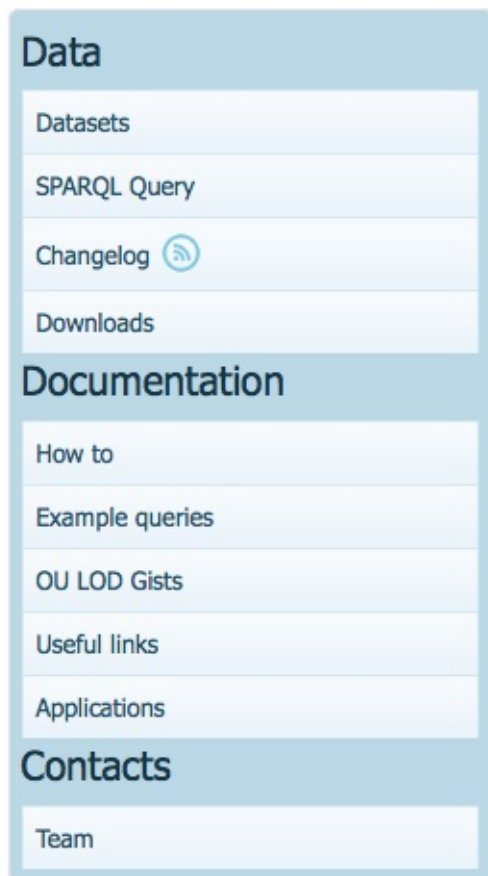
"[The Open University](#)" (La Universidad Abierta) es una universidad de Gran Bretaña que imparte sus estudios a distancia. Tiene una filosofía de apertura a las personas, métodos e ideas. La mayoría de sus cursos no tienen requisitos formales de acceso y ofrecen una gran cantidad de material abierto para su consulta de forma gratuita.



Proponemos este caso de aplicación, ya que la universidad proporciona una [página web](#) en la que podemos realizar consultas a su base de datos construida bajo los principios de Linked Data. En la portada nos encontramos con la opción de echar un vistazo rápido a entidades de las que ya conozcamos su identificador, el acceso a los datasets y la licencia Creative Commons de los datos, junto con la imagen en la que reconocen que cumplen con las 5 estrellas del modelo.

A la derecha encontramos un menú con las siguientes opciones:

- Datasets. Podremos acceder al listado de todos los datasets.
- SPARQL Query. Para realizar consultas SPARQL en los datos.
- Changelog. Novedades introducidas en la plataforma a lo largo del tiempo.
- Downloads. Descarga de datasets en formato RDF/XML.
- Documentation. Guías de cómo usar la página y consultas SPARQL de ejemplo.
- Contacts. Equipo encargado del mantenimiento de la página.



## Ejemplos de consultas SPARQL

### Cursos de español disponibles en Alemania

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT ?c ?l WHERE {
  ?c <http://purl.org/dc/terms/subject> <http://data.open.ac.uk/topic/spanish> .
  ?c rdfs:label ?l .
  ?c <http://data.open.ac.uk/saou/ontology#isAvailableIn> <http://sws.geonames.org/2921044/> }

```

Podemos ver como en topic seleccionan "spanish", que define la materia, y la última línea indica que los cursos estén disponibles en Alemania a través de un URI de localización.

### Cualquier material que contenga la palabra "learning"

```
PREFIX mlo: <http://purl.org/net/mlo/>
SELECT ?thing ?description
FROM <http://data.open.ac.uk/context/openlearn>
FROM <http://data.open.ac.uk/context/podcast>
where {
    ?thing <http://purl.org/dc/terms/description> ?description .
    FILTER EXISTS {
        { ?thing a <http://data.open.ac.uk/openlearn/ontology/OpenLearnUnit> }
        UNION
        { ?thing a <http://data.open.ac.uk/podcast/ontology/VideoPodcast> }
    } .
    FILTER regex(str(?description), "learning", "i" )
}
```

Busca el término "learning" (especificado en la última línea) en cualquier podcast y material openlearn.

# De Data a Linked Data

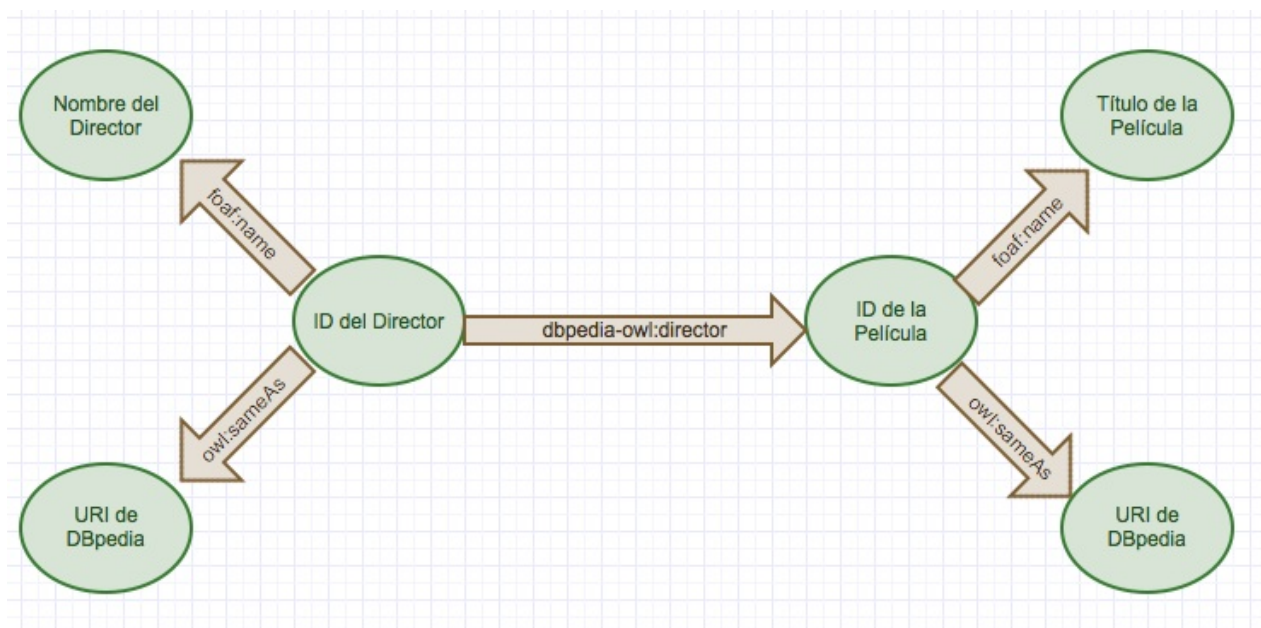
A continuación vamos a seguir un tutorial paso a paso para convertir un archivo de datos de una hoja de cálculo en datos relacionados como Linked Data.

## 1. Fichero de datos de partida

Directores		Películas		Relaciones	
ID	Nombre	ID	Título	DirectorID	TítuloID
1	Álex de la Iglesia	1	El otro lado de la cama	3	7
2	Benito Zambrano	2	Mientras duermes	5	2
3	Emilio Martínez-Lázaro	3	La voz dormida	2	3
4	Juan Antonio Bayona	4	El día de la bestia	1	4
5	Jaume Balagueró	5	Lo imposible	4	5
		6	Frágiles	5	6
		7	Ocho apellidos vascos	3	7
		8	Balada triste de trompeta	1	8
		9	El orfanato	4	9
		10	Solas	2	10

En la imagen tenemos los datos a modelar. El archivo lo podéis descargar [aquí](#). Podemos ver como tenemos a una serie de directores y películas, todos identificados con un ID y otra columna en la que tenemos las relaciones entre ellos. Nuestro objetivo es crear un enlace semántico para que una máquina pueda conocer las relaciones y enlazarlas con entidades de fuentes externas para convertir la información en Linked Data.

## 2. Diagrama de entidades



En este diagrama podemos ver cómo pretendemos relacionar los datos. Cada ID del Director estará relacionado con su nombre, los ID de las películas que ha dirigido y la URI que se refiere a la misma persona en DBpedia. Por su parte, cada ID de Película estará relacionado con su título y con la URI que se refiere a la película en DBpedia.

### 3. Preparación del entorno



Para lograr nuestro objetivo, utilizaremos [Google Refine](#). Ésta es una poderosa herramienta para limpiar y unificar nuestros datos en distintos formatos. Para poder exportarlos a RDF, necesitaremos instalar una extensión llamada [RDF Refine](#). Primero descargaremos e instalaremos Google Refine desde su página de [descargas](#) (elegiremos la correspondiente a nuestro sistema operativo). Luego, procederemos a descargar RDF Refine. Extraemos la carpeta que viene comprimida y la movemos a la carpeta "extensions" situada en el directorio de instalación de Google Refine. Con esto ya tendremos preparado el entorno con el que vamos a trabajar.

Abrimos Google Refine. Comprobamos que estén marcados "Create Project" y "This Computer" y hacemos click en examinar para buscar nuestro archivo con los datos. Después pulsamos en el botón "Next".

### 4. Creando el proyecto

	Column 1	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Column 8
1.	Directores			Películas			Relaciones	
2.								
3.	ID	Nombre		ID	Título		DirectorID	TítuloID
4.	1	Álex de la Iglesia		1	El otro lado de la cama		3	7
5.	2	Benito Zambrano		2	Mientras duermes		5	2
6.	3	Emilio Martínez-Lázaro		3	La voz dormida		2	3
7.	4	Juan Antonio Bayona		4	El día de la bestia		1	4
8.	5	Jaume Balagueró		5	Lo imposible		4	5
9.				6	Frágiles		5	6
10.				7	Ocho apellidos vascos		3	7
11.				8	Balada triste de trompeta		1	8
12.				9	El orfanato		4	9
13.				10	Solas		2	10

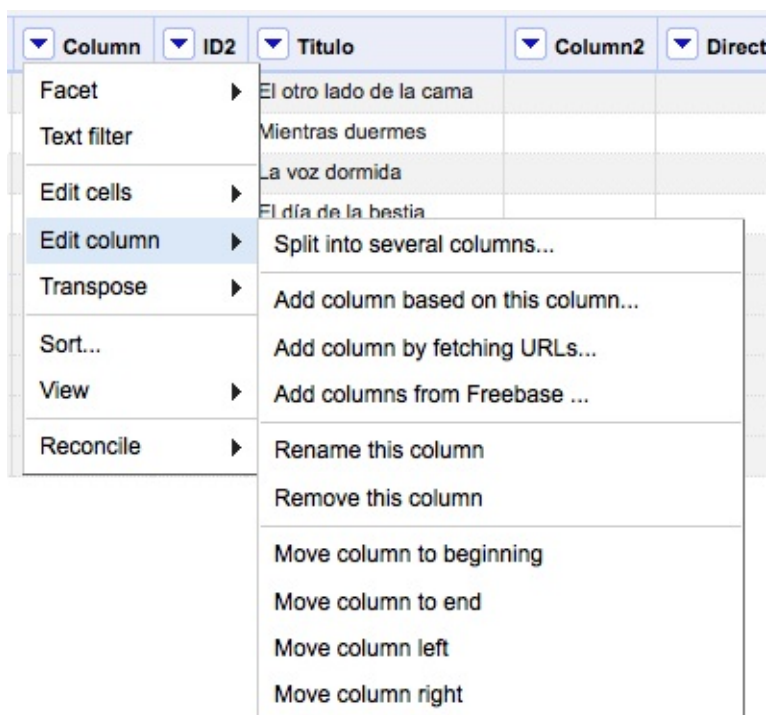
Indicamos que ignore las tres primeras líneas que no contienen datos. También que no almacene las filas ni las celdas en blanco, tal y como aparece en la siguiente imagen.

- ☒ Ignore first  
☒ Parse next  
☐ Discard initial  
☐ Load at most
- 3 line(s) at beginning of file  
 1 line(s) as column headers  
 0 row(s) of data  
 0 row(s) of data
- ☐ Store blank rows  
☐ Store blank cells as nulls  
☐ Store file source (file names, URLs) in each row

Por último, pulsamos en "Create Project" arriba a la derecha.

## 5. Preparando los datos

Si hacemos click en la flechita al lado del nombre de cada columna, podremos eliminar las columnas que están en blanco.



También podremos cambiar el nombre de la columna que queramos para clarificar.

▼ All	▼ ID	▼ Nombre Director	▼ ID2	▼ Título Película	▼ DirectorID	▼ TítuloID
★	1.	1	1	Álex de la Iglesia	3	7
★	2.	2	2	Benito Zambrano	5	2
★	3.	3	3	Emilio Martínez-Lázaro	2	3
★	4.	4	4	Juan Antonio Bayona	1	4
★	5.	5	5	Jaume Balagueró	4	5
★			6	Frágiles	5	6
★			7	Ocho apellidos vascos	3	7
★			8	Balada triste de trompeta	1	8
★			9	El orfanato	4	9
★			10	Solas	2	10



## 6. Enlazando con DBpedia

Para enlazar los datos con fuentes externas como DBpedia, necesitamos añadir un servicio de conciliación. Esto será un enlace a una web que nos permita realizar consultas SPARQL en los datos que queremos conciliar. En el caso de DBpedia en español será <http://es.dbpedia.org/sparql>.

### Virtuoso SPARQL Query Editor

Default Data Set Name (Graph IRI)

Query Text

```
select distinct ?Concept where {[ ] a ?Concept} LIMIT 100
```

(Security restrictions of this server do not allow you to retrieve remote RDF data, see [details](#).)

Results Format:

HTML

Execution timeout:

0

milliseconds (values less than 1000 are ignored)

Options:

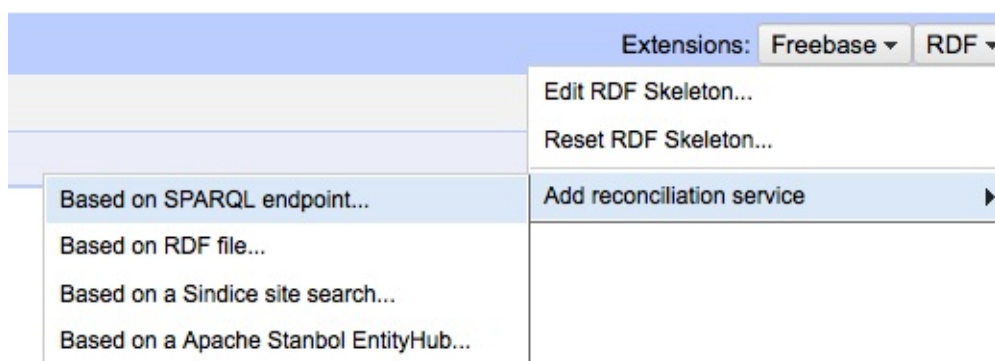
☒ Strict checking of void variables

(The result can only be sent back to browser, not saved on the server, see [details](#))

Run Query

Reset

Para añadirlo, pulsamos en el botón RDF arriba a la derecha, "Add reconciliation service" y "Based on SPARQL endpoint".



Le ponemos un nombre (DBpedia), le añadimos la URL a la web de consulta SPARQL y elegimos como tipo "Virtuoso".

### Add SPARQL-based reconciliation service

**Name:**   
A human readable name

**Endpoint details**

**Endpoint URL:**

**Graph URI:**   
Leave empty to use the default graph

**Type:**  This determines the syntax that will be used for search

**Label properties**  
 Select properties that are used to label resources in the endpoint. These properties will be used to match resources:

☒ `rdfs:label`
☐ `skos:prefLabel`
☐ `dcterms:title`
☐ `dc:title`  
☐ `foaf:name`  
☐ `Other...`

Ahora en la columna "Nombre Director", en el menú desplegable elegimos "Reconcile" y "Start reconciling..." para comenzar a relacionar los datos con las URI de DBpedia.

▼ Nombre Director	▼ ID2	▼ Título Película	▼
Facet	1	El otro lado de la cama	
Text filter	2	Mientras duermes	
	3	La voz dormida	
Edit cells	4	El día de la bestia	
Edit column	5	Lo imposible	
Transpose	6	Frágiles	
	7	Ocho apellidos vascos	
Sort...	8	Balada triste de trompeta	
View	9	El orfanato	
Reconcile	Start reconciling... Facets QA facets Actions Copy reconciliation data... Discover related RDF datasets...		

Elegimos que busque similitudes con entidades de tipo "foaf:Person" y pulsamos "Start Reconciling".

#### Reconcile column "Nombre Director"

[» Access Service API](#)

Freebase Query-based Reconciliation ✕

Sindice ✕

**DBpedia** ✕

Reconcile each cell to an entity of one of these types:

- ☐ skos:Concept  
http://www.w3.org/2004/02/skos/core#Concept
- ☐ owl:Thing  
http://www.w3.org/2002/07/owl#Thing
- ☐ http://wikidata.dbpedia.org/resource/Q215627
- ☐ schema:Person  
http://schema.org/Person
- ☒ foaf:Person  
http://xmlns.com/foaf/0.1/Person
- ☐ http://wikidata.dbpedia.org/resource/Q5
- ☐ http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Agent
- ☐ http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#NaturalPerson
- ☐ dbo:Actor  
http://dbpedia.org/ontology/Actor
- ☐ dbo:Agent

☐ Reconcile against type:

☐ Reconcile against no particular type

☒ Auto-match candidates with high confidence

Also use relevant details from other columns:

Column	Include?	As Property
ID	<input type="checkbox"/>	<input type="text"/>
ID2	<input type="checkbox"/>	<input type="text"/>
Título Película	<input type="checkbox"/>	<input type="text"/>
DirectorID	<input type="checkbox"/>	<input type="text"/>
TítuloID	<input type="checkbox"/>	<input type="text"/>

Add Standard Service...
Add Namespaced Service...
Start Reconciling
Cancel

Como vemos, prácticamente no habrá encontrado ninguna coincidencia. Con la DBpedia en inglés es más eficaz, pero la española no recoge bien los acentos por ejemplo. Así que le ayudaremos a relacionar los datos.

<input type="checkbox"/> All	<input type="checkbox"/> ID	<input type="checkbox"/> Nombre Director
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
1.	1	<b>Álex de la Iglesia</b> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <a href="#">Álex de la Iglesia (1)</a> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <a href="#">Álex de la Iglesia (1)</a> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <a href="#">Create new topic</a> <a href="#">Search for match</a>
2.	2	<b>Benito Zambrano</b> <a href="#">Choose new match</a>
3.	3	<b>Emilio Martínez-Lázaro</b> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <a href="#">Create new topic</a> <a href="#">Search for match</a>
4.	4	<b>Juan Antonio Bayona</b> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <a href="#">Juan Antonio Bayona (1)</a> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <a href="#">Juan Antonio Bayona (1)</a> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <a href="#">Create new topic</a> <a href="#">Search for match</a>
5.	5	<b>Jaume Balagueró</b> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <a href="#">Jaume Balagueró (1)</a> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <a href="#">Jaume Balagueró (1)</a> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <a href="#">Create new topic</a> <a href="#">Search for match</a>
<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	<input type="checkbox"/>	

Pulsamos en "Search for match" para cada hallazgo (porque si elegimos alguno de los que nos vienen, probablemente enlazará a una página en blanco) y buscamos el nombre del director sin acentos. Elegiremos el enlace que aparecerá de la DBpedia española (empieza por "es").

**Search for Match**  
 Search for "Emilio Martínez-Lázaro"  
☒ Match other cells with same content  
☐ Match this cell only

Emilio Martínez-Lázaro  
 Select an item from the list:  
 Emilio Martinez Lazaro  
**Emilio Martinez-Lazaro**

[http://es.dbpedia.org/resource/Emilio\\_Martinez-Lazaro](http://es.dbpedia.org/resource/Emilio_Martinez-Lazaro)

Una vez los tengamos, es posible pulsar en cada nombre para comprobar que nos lleva a la página correcta de la DBpedia en español.

<input type="checkbox"/> Nombre Director
<a href="#">Alex de la Iglesia</a> <a href="#">Choose new match</a>
<a href="#">Benito Zambrano</a> <a href="#">Choose new match</a>
<a href="#">Emilio Martinez-Lazaro</a> <a href="#">Choose new match</a>
<a href="#">Juan Antonio Bayona</a> <a href="#">Choose new match</a>
<a href="#">Jaume Balagueró</a> <a href="#">Choose new match</a>

Ahora añadiremos una nueva columna con los enlaces de las entidades a la DBPedia. Para ello, pulsamos en la flecha al lado de "Nombre Director" y seleccionamos "Edit column" -> "Add column based on this column...".

Nombre Director	ID2	Titulo Película	DirectorID
Facet	1	El otro lado de la cama	3
Text filter	2	Mientras duermes	5
Edit cells			
Edit column	Split into several columns...		
Transpose	Add column based on this column...		
Sort...	Add column by fetching URLs...		
View	Add columns from Freebase ...		
Reconcile	Rename this column		
	Remove this column		
	Move column to beginning		
	Move column to end		
	Move column left		
	Move column right		

Le ponemos el nombre que queramos y en expresión escribiremos "cell.recon.match.id" para que reconozca los enlaces de cada director. Esto es una expresión [GREL](#), que hace referencia al valor reconciliado de la celda en cuestión. En las expresiones GREL pueden emplearse diversas [variables](#). Ya podemos pulsar en "OK".

### Add column based on column Nombre Director

New column name

On error ☒ set to blank ☐ store error ☐ copy value from original column

Expression  Language  No syntax error.

**Preview** History Starred Help

row	value	cell.recon.match.id
1.	Álex de la Iglesia	<a href="http://es.dbpedia.org/resource/Alex_de_la_Iglesia">http://es.dbpedia.org/resource/Alex_de_la_Iglesia</a>
2.	Benito Zambrano	<a href="http://es.dbpedia.org/resource/Benito_Zambrano">http://es.dbpedia.org/resource/Benito_Zambrano</a>
3.	Emilio Martínez-Lázaro	<a href="http://es.dbpedia.org/resource/Emilio_Martinez-Lazaro">http://es.dbpedia.org/resource/Emilio_Martinez-Lazaro</a>
4.	Juan Antonio Bayona	<a href="http://es.dbpedia.org/resource/Juan_Antonio_Bayona">http://es.dbpedia.org/resource/Juan_Antonio_Bayona</a>
5.	Jaume Balagueró	<a href="http://es.dbpedia.org/resource/Jaume_Balaguero">http://es.dbpedia.org/resource/Jaume_Balaguero</a>
6.		Error: Cannot retrieve field from null

OK Cancel

Como vemos en la imagen, nos aparecerá la nueva columna.

▼ Nombre Director	▼ URI DBpedia del Director
Alex de la Iglesia <a href="#">Choose new match</a>	<a href="http://es.dbpedia.org/resource/Alex_de_la_Iglesia">http://es.dbpedia.org/resource/Alex_de_la_Iglesia</a>
Benito Zambrano <a href="#">Choose new match</a>	<a href="http://es.dbpedia.org/resource/Benito_Zambrano">http://es.dbpedia.org/resource/Benito_Zambrano</a>
Emilio Martinez-Lazaro <a href="#">Choose new match</a>	<a href="http://es.dbpedia.org/resource/Emilio_Martinez-Lazaro">http://es.dbpedia.org/resource/Emilio_Martinez-Lazaro</a>
Juan Antonio Bayona <a href="#">Choose new match</a>	<a href="http://es.dbpedia.org/resource/Juan_Antonio_Bayona">http://es.dbpedia.org/resource/Juan_Antonio_Bayona</a>
Jaume Balaguero <a href="#">Choose new match</a>	<a href="http://es.dbpedia.org/resource/Jaume_Balaguero">http://es.dbpedia.org/resource/Jaume_Balaguero</a>

Ahora haremos lo mismo con las películas, con la columna "Titulo Película". En este caso, elegiremos la opción "Reconcile against no particular type", ya que no reconoce la ontología "Film" de la DBpedia en español, sólo la que está en inglés.



### Reconcile column "Titulo Película"

Freebase Query-based Reconciliation » Access Service API

Sindice ×

**DBpedia** ×

Reconcile each cell to an entity of one of these types: Also use relevant details from other columns:

- ☐ schema:CreativeWork  
<http://schema.org/CreativeWork>
- ☐ schema:Movie  
<http://schema.org/Movie>
- ☐ <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#InformationEntity>
- ☐ dbo:Film  
<http://dbpedia.org/ontology/Film>
- ☐ dbo:Wikidata:Q11424  
<http://dbpedia.org/ontology/Wikidata:Q11424>
- ☐ dbo:Work  
<http://dbpedia.org/ontology/Work>
- ☐ dbo:TelevisionShow  
<http://dbpedia.org/ontology/TelevisionShow>
- ☐ schema:TVEpisode  
<http://schema.org/TVEpisode>
- ☐ dbo:TelevisionEpisode  
<http://dbpedia.org/ontology/TelevisionEpisode>

☐ Reconcile against type:

☒ Reconcile against no particular type

☒ Auto-match candidates with high confidence

Column	Include?	As Property
ID	<input type="checkbox"/>	<input type="text"/>
Nombre	<input type="checkbox"/>	<input type="text"/>
Director	<input type="checkbox"/>	<input type="text"/>
URI	<input type="checkbox"/>	<input type="text"/>
DBpedia del	<input type="checkbox"/>	<input type="text"/>
Director	<input type="checkbox"/>	<input type="text"/>
ID2	<input type="checkbox"/>	<input type="text"/>
DirectorID	<input type="checkbox"/>	<input type="text"/>
TituloID	<input type="checkbox"/>	<input type="text"/>

Add Standard Service... Add Namespaced Service... Start Reconciling Cancel

Como en el caso de los directores, podremos conciliar los datos manualmente. Buscaremos el nombre de la película en español, sin acentos. También comprobaremos que nos lleve a los enlaces correctos de la DBpedia en español.

### Search for Match

Search for "El otro lado de la cama"

☒ Match other cells with same content

☐ Match this cell only

El otro lado de la cama

Select an item from the list:

- El otro lado de la cama
- El otro lado de la cama

Match New Topic Don't Reconcile Cell Cancel

[http://es.dbpedia.org/resource/El\\_otro\\_lado\\_de\\_la\\_cama](http://es.dbpedia.org/resource/El_otro_lado_de_la_cama)

Una vez tengamos todas las películas enlazadas tendremos algo como lo que se muestra en la siguiente imagen.

▼ Título Película
El otro lado de la cama Choose new match
Mientras duermes Choose new match
La voz dormida Choose new match
El día de la Bestia Choose new match
Lo imposible Choose new match
Fragiles (película) Choose new match
Ocho apellidos vascos Choose new match
Balada triste de trompeta Choose new match
El orfanato Choose new match
Solas (película) Choose new match

Creamos ahora la nueva columna con las URI como ya hicimos con "Nombre Director".

▼ Título Película	▼ URI DBpedia de la Película
El otro lado de la cama Choose new match	<a href="http://es.dbpedia.org/resource/El_otro_lado_de_la_cama">http://es.dbpedia.org/resource/El_otro_lado_de_la_cama</a>
Mientras duermes <a href="#">edit</a> Choose new match	<a href="http://es.dbpedia.org/resource/Mientras_duermes">http://es.dbpedia.org/resource/Mientras_duermes</a>
La voz dormida Choose new match	<a href="http://es.dbpedia.org/resource/La_voz_dormida">http://es.dbpedia.org/resource/La_voz_dormida</a>
El día de la Bestia Choose new match	<a href="http://es.dbpedia.org/resource/El_dia_de_la_Bestia">http://es.dbpedia.org/resource/El_dia_de_la_Bestia</a>
Lo imposible Choose new match	<a href="http://es.dbpedia.org/resource/Lo_imposible">http://es.dbpedia.org/resource/Lo_imposible</a>
Fragiles (película) Choose new match	<a href="http://es.dbpedia.org/resource/Fragiles_(película)">http://es.dbpedia.org/resource/Fragiles_(película)</a>
Ocho apellidos vascos Choose new match	<a href="http://es.dbpedia.org/resource/Ocho_apellidos_vascos">http://es.dbpedia.org/resource/Ocho_apellidos_vascos</a>
Balada triste de trompeta Choose new match	<a href="http://es.dbpedia.org/resource/Balada_triste_de_trompeta">http://es.dbpedia.org/resource/Balada_triste_de_trompeta</a>
El orfanato Choose new match	<a href="http://es.dbpedia.org/resource/El_orfanato">http://es.dbpedia.org/resource/El_orfanato</a>
Solas (película) Choose new match	<a href="http://es.dbpedia.org/resource/Solas_(película)">http://es.dbpedia.org/resource/Solas_(película)</a>

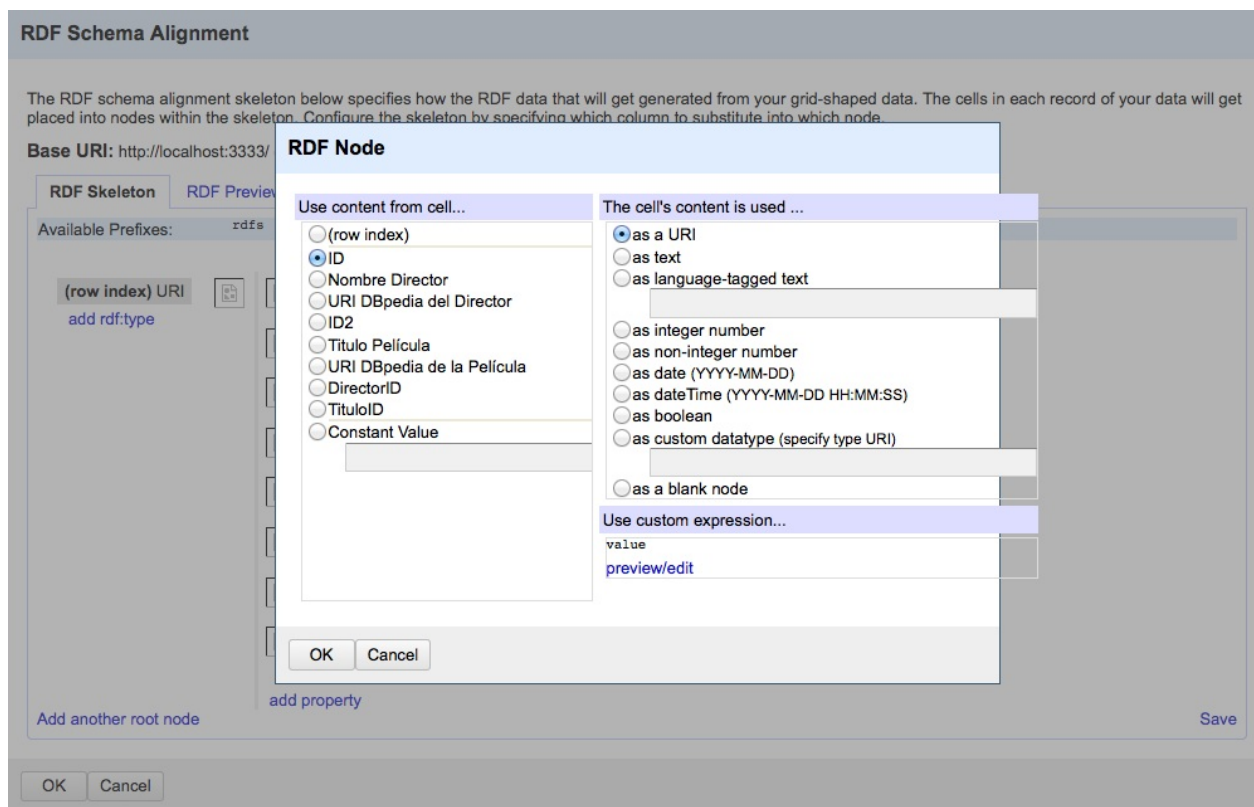
## 7. Completando y exportando el RDF

A continuación editaremos el esqueleto del RDF para indicar la relación entre nuestras columnas. Para ello, vamos al menú "RDF" arriba a la derecha y pulsamos en "Edit RDF Skeleton".





Pulsamos en "(row index) URI" y elegimos "ID", que es el ID de los directores.



En "add rdf:type" le añadimos el tipo "foaf:person".



A continuación estableceremos las propiedades. La primera será "foaf:name" que apuntará a las celdas de "Nombre Director". Así enlazamos cada ID con su nombre.

The screenshot shows a table editor interface. On the left, there's a sidebar with 'ID URI' and 'foaf:Person' (with a blue 'X' icon) and a link 'add rdf:type'. The main area shows a table with columns: 'ID URI', 'foaf:Person', and several property columns. The 'URI DBpedia del Director' cell is highlighted. Below the table, there's a link 'add property'.

Pulsamos en "URI DBpedia del Director cell" y arriba a la derecha indicamos que el contenido es "as a URI" para que lo trate como tal.

The 'RDF Node' dialog box is shown. It has two main sections: 'Use content from cell...' and 'The cell's content is used ...'. In the first section, 'URI DBpedia del Director' is selected. In the second section, 'as a URI' is selected. There are also options for 'as text', 'as language-tagged text', 'as integer number', 'as non-integer number', 'as date (YYYY-MM-DD)', 'as dateTime (YYYY-MM-DD HH:MM:SS)', 'as boolean', 'as custom datatype (specify type URI)', and 'as a blank node'. At the bottom, there are 'OK' and 'Cancel' buttons.

Le ponemos la propiedad "owl:sameAs". Así le estamos indicando que nuestro director es el mismo que el que aparece en esa URI de la DBpedia. Borramos el resto de propiedades sin definir. Nos quedará algo como lo que vemos en la siguiente imagen.

**ID URI** ☐ [foaf:Person](#) [add rdf:type](#)

[add property](#)

**Nombre Director** cell

[foaf:name](#) →

[owl:sameAs](#) → **URI DBpedia del Director URI** ☐ [add rdf:type](#)

A continuación pulsamos en "Add another root node" abajo a la izquierda y seleccionamos "ID2" para hacer lo propio con las películas.

**RDF Node**

Use content from cell...

- ☐ (row index)
- ☐ ID
- ☐ Nombre Director
- ☐ URI DBpedia del Director
- ☒ ID2
- ☐ Título Película
- ☐ URI DBpedia de la Película
- ☐ DirectorID
- ☐ TítuloID
- ☐ Constant Value

The cell's content is used ...

- ☒ as a URI
- ☐ as text
- ☐ as language-tagged text
- ☐ as integer number
- ☐ as non-integer number
- ☐ as date (YYYY-MM-DD)
- ☐ as dateTime (YYYY-MM-DD HH:MM:SS)
- ☐ as boolean
- ☐ as custom datatype (specify type URI)
- ☐ as a blank node

Use custom expression...

value

[preview/edit](#)

OK Cancel

Le ponemos la propiedad "foaf:name" apuntando a la columna "Título Película" y la propiedad "owl:sameAs" apuntando a la columna con las URI de DBpedia de forma similar a los directores.

**ID2 URI** ☐ [add rdf:type](#)

[add property](#)

**Título Película** cell

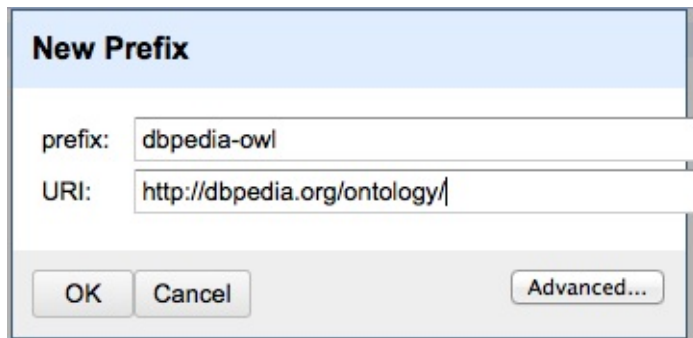
[foaf:name](#) →

[owl:sameAs](#) → **URI DBpedia de la Película URI** ☐ [add rdf:type](#)

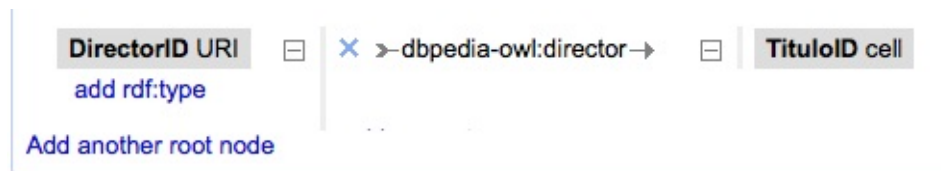
Añadiremos un último "root node" para definir la relación entre los directores y las películas. Seleccionamos "DirectorID" con una propiedad que apuntará a "TituloID".



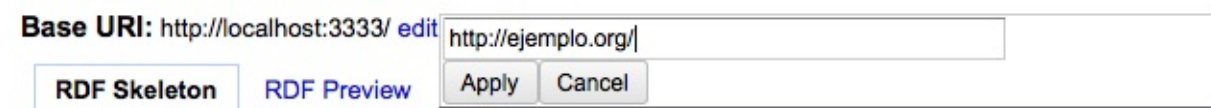
La propiedad que nos interesa es "dbpedia-owl:director" que no está definida por defecto, así que tendremos que incluir ese vocabulario. Para ello pulsamos en "+add prefix" y ponemos en prefix "dbpedia-owl" y en URI "<http://dbpedia.org/ontology/>". Pulsamos en "OK" y añadirá ese vocabulario para que podamos usarlo.



Ahora sí podremos definir la propiedad "dbpedia-owl:director" como indicamos en el diagrama al principio.



A continuación editaremos la "Base URI" para referenciar los distintos elementos de nuestros datos de una forma legible. Así que pulsamos en "edit" y escribimos la que queramos tomar como base, en nuestro ejemplo "<http://ejemplo.org/>". Es importante poner la barra "/" al final, porque si no, no quedarán bien referenciados los enlaces.



Para que cada entidad siga un patrón de referencia en la URI, tendremos que configurarlo. Para ello, pulsaremos primero en "ID URI".

ID URI

foaf:Person

dbpedia-owl:MovieDirector

add rdf:type

ID2 URI

dbpedia-owl:Film

add rdf:type

Editaremos el apartado en el que pone "value", ya que esto sólo pondrá en la URI el ID. En "Expression" podríamos poner "'director/' + value". Esto indicaría en cada URI que se trata de un director y añadiría su ID. El problema es que nos pone algunas filas vacías. Para solucionarlo, pondremos una condición de que sólo haga esto en las filas que no estén vacías: "if (value == '', null, 'director/' + value)".

Preview URI values

Expression

Language Google Refine Expression Language (GREL)

if(value=='', null, 'director/' + value)

No syntax error.

Preview

History

Starred

Help

row	value	if(value=='', null, 'director/' + value)	resolved against the base URI
1.	1	director/1	http://ejemplo.org/director/1
2.	2	director/2	http://ejemplo.org/director/2
3.	3	director/3	http://ejemplo.org/director/3
4.	4	director/4	http://ejemplo.org/director/4
5.	5	director/5	http://ejemplo.org/director/5
6.		null	null
7		null	null

OK

Cancel

Tendremos ahora que hacer lo mismo con "ID2 URI" para las películas. Esta vez si bastará con poner en la expresión "'film/' + value".

### Preview URI values

Expression
Language
Google Refine Expression Language (GREL)

'film/' + value
No syntax error.

Preview
History
Starred
Help

row	value	'film/' + value	resolved against the base URI
1.	1	film/1	http://ejemplo.org/film/1
2.	2	film/2	http://ejemplo.org/film/2
3.	3	film/3	http://ejemplo.org/film/3
4.	4	film/4	http://ejemplo.org/film/4
5.	5	film/5	http://ejemplo.org/film/5
6.	6	film/6	http://ejemplo.org/film/6
7	7	film/7	http://ejemplo.org/film/7

OK
Cancel

No debemos olvidarnos de "DirectorID" y "TituloID".

DirectorID URI
add rdf:type
Add another root node

X
dbpedia-owl:director

TituloID cell

En "DirectorID" podremos poner ahora "'director/' + value" sin problemas.



### Preview URI values

Expression
Language
Google Refine Expression Language (GREL)

'director/' + value
No syntax error.

Preview
History
Starred
Help

row	value	'director/' + value	resolved against the base URI
1.	3	director/3	http://ejemplo.org/director/3
2.	5	director/5	http://ejemplo.org/director/5
3.	2	director/2	http://ejemplo.org/director/2
4.	1	director/1	http://ejemplo.org/director/1
5.	4	director/4	http://ejemplo.org/director/4
6.	5	director/5	http://ejemplo.org/director/5
7.	3	director/3	http://ejemplo.org/director/3

OK
Cancel

En "TituloID" pondremos lo mismo que en "ID2".

### Preview URI values

Expression
Language
Google Refine Expression Language (GREL)

'film/' + value
No syntax error.

Preview
History
Starred
Help

row	value	'film/' + value
1.	7	film/7
2.	2	film/2
3.	3	film/3
4.	4	film/4
5.	5	film/5
6.	6	film/6
7.	7	film/7

OK
Cancel

Si nos damos cuenta, "TituloID" no está configurado como URI, así que tendremos que cambiarlo. Pulsamos en "TituloID" y seleccionamos "URI" arriba a la derecha.

<b>DirectorID URI</b> add rdf:type Add another root node	X dbpedia-owl:director →	<b>TituloID URI</b> add rdf:type
--	--------------------------	-------------------------------------

Ya tendremos nuestro RDF listo. Podremos ver cómo quedará en la pestaña "RDF Preview".

**RDF Schema Alignment**

The RDF schema alignment skeleton below specifies how the RDF data that will get generated from your grid-shaped data. The cells in each record of your data will get placed into nodes within the skeleton. Configure the skeleton by specifying which column to substitute into which node.

**Base URI:** <http://ejemplo.org/> [edit](#)

[RDF Skeleton](#)
[RDF Preview](#)

This is a sample *Turtle* representation of (up-to) the *first 10 rows*

```

@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix dbpedia-owl: <http://dbpedia.org/ontology/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

<http://ejemplo.org/director/1> a foaf:Person , dbpedia-owl:MovieDirector ;
    foaf:name "Álex de la Iglesia" ;
    owl:sameAs <http://es.dbpedia.org/resource/Alex_de_la_Iglesia> .

<http://ejemplo.org/film/1> a dbpedia-owl:Film ;
    foaf:name "El otro lado de la cama" ;
    owl:sameAs <http://es.dbpedia.org/resource/El_otro_lado_de_la_cama> .

<http://ejemplo.org/director/3> dbpedia-owl:director <http://ejemplo.org/film/7> .

<http://ejemplo.org/director/2> a foaf:Person , dbpedia-owl:MovieDirector ;
    foaf:name "Benito Zambrano" ;
    owl:sameAs <http://es.dbpedia.org/resource/Benito_Zambrano> .

<http://ejemplo.org/film/2> a dbpedia-owl:Film ;
    foaf:name "Mientras duermes" ;
    owl:sameAs <http://es.dbpedia.org/resource/Mientras_duermes> .

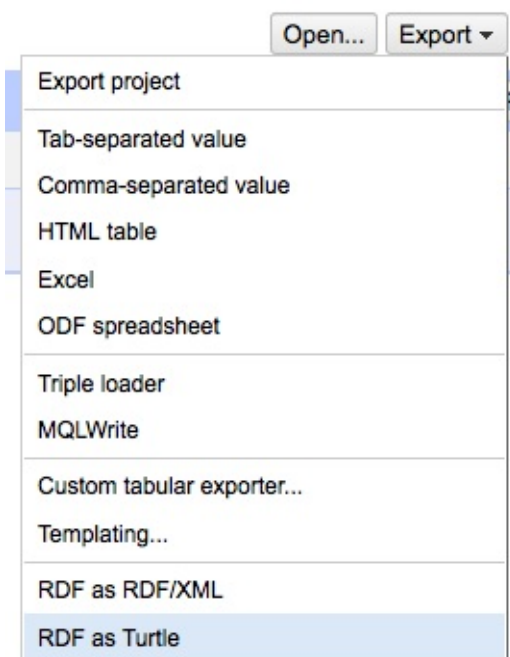
<http://ejemplo.org/director/5> dbpedia-owl:director <http://ejemplo.org/film/2> .

<http://ejemplo.org/director/3> a foaf:Person , dbpedia-owl:MovieDirector ;
    foaf:name "Emilio Martínez-Lázaro" ;
    owl:sameAs <http://es.dbpedia.org/resource/Emilio_Martinez-Lazaro> .
  
```

OK
 Cancel

Por último, podremos exportarlo en el menú "Export" arriba a la derecha en formato Turtle o RDF/XML.





Con esto termina el tutorial, ya hemos convertido nuestros datos en Linked Data.

[Enlace](#) al fichero con formato Turtle. Haz click derecho en el mismo y pulsa en "Guardar enlace como".

Fuente: <https://www.youtube.com/watch?v=XdpzmGxA33U>

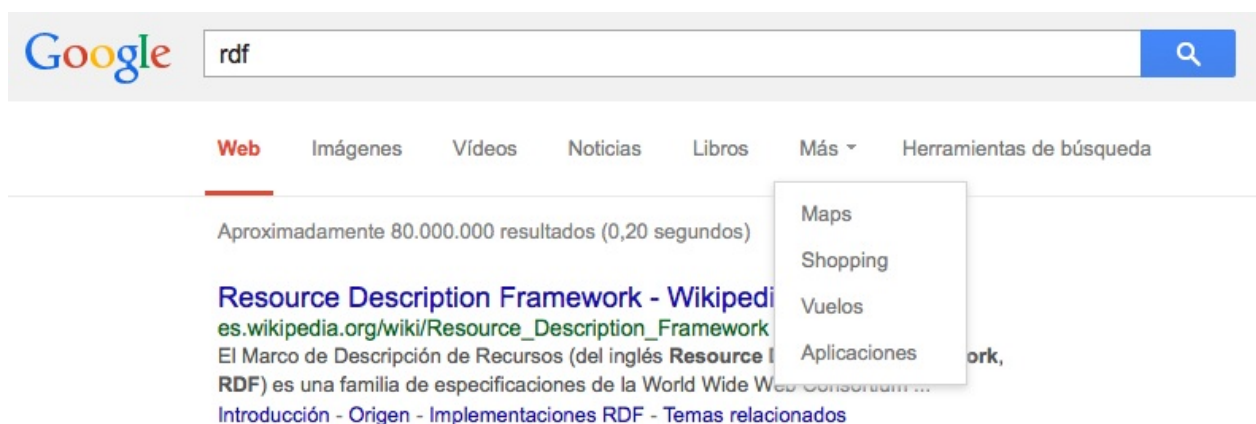
## Ejemplo de microformato

Según el W3C, los microformatos son "conjuntos de formatos de datos abiertos y simples, desarrollados sobre estándares ya existentes, ampliamente adoptados, incluyendo XHTML (HTML y XML) y CSS.". Se utilizan para incluir fichas de contenido semántico sobre distintas cuestiones, ya sean eventos, localizaciones, tarjetas de visita... etc.

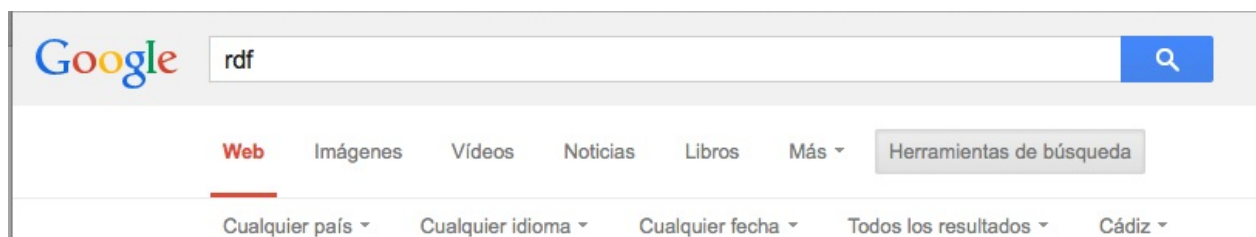
En este capítulo veremos cómo pueden ser útiles los microformatos con el ejemplo de Google Recipe y cómo utilizar la herramienta "Structured Data Testing Tool" de Google para comprobar si una página está anotada con microformatos, microdatos o RDFa.

## Google Recipe

Si realizamos cualquier búsqueda en [Google](#), veremos las opciones de búsqueda que tenemos. Podemos ver en la imagen cómo podemos buscar en las páginas web, en imágenes, en vídeos, en noticias, en libros, en mapas, en compras, en vuelos o aplicaciones.



Además de eso, podemos filtrar las búsquedas por país, idioma, fecha o ubicación.



Sin embargo, ninguna de estas opciones utiliza la semántica. La mayoría buscan en el contenido de las páginas web, los libros, los títulos y descripciones de vídeos e imágenes... La fecha y la ubicación también quedan almacenadas en el servidor.

A continuación, visitaremos la página que muestra las funcionalidades de [Google Recipes](#). Aquí podremos ver un vídeo y una serie de descripciones que nos explican que podemos realizar una búsqueda de recetas filtrando por ingredientes, tiempo de preparación o calorías. Esto es posible gracias a que la página web que contiene la receta debe contener a su vez un microformato con un vocabulario concreto en el que queden reflejados esos datos.

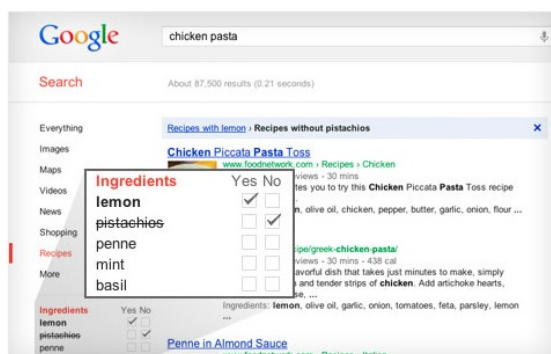
## Encuentra recetas en la Web

Una vez que hayas buscado una receta o unos ingredientes específicos, selecciona la opción **Recetas** en el panel de la izquierda de la página de resultados de búsqueda.

Puedes filtrar los resultados en función de los **ingredientes**, del **tiempo de preparación** o del número de **calorías**. (Ten en cuenta que la vista de recetas solo está disponible en inglés).

[Probar la vista de recetas ahora](#)

[Más información para autores de recetas](#)



### Restringe la búsqueda

Si buscas una receta o un ingrediente en Google, obtendrás una gran cantidad de resultados, pero no todos serán de recetas. Ahora puedes restringir los resultados de búsqueda para que solo se muestren recetas.

### Información de recetas a un clic

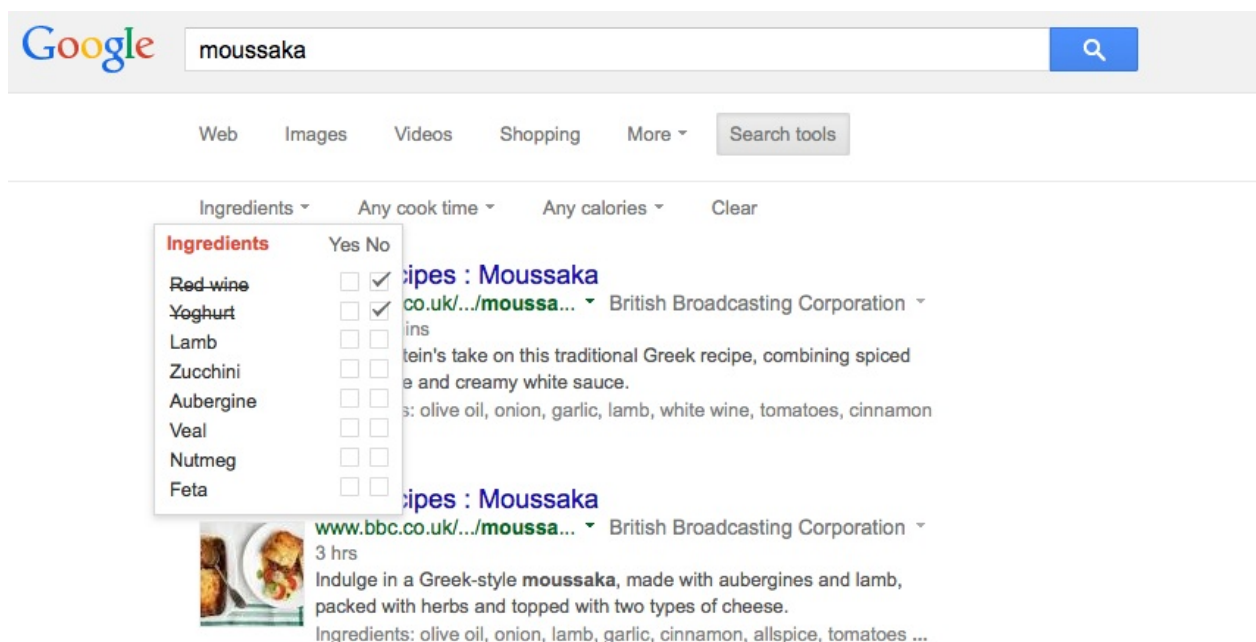
La página de resultados de búsqueda incluye valoraciones, ingredientes e imágenes que te ayudarán a encontrar la receta adecuada.

### Personalización de los resultados

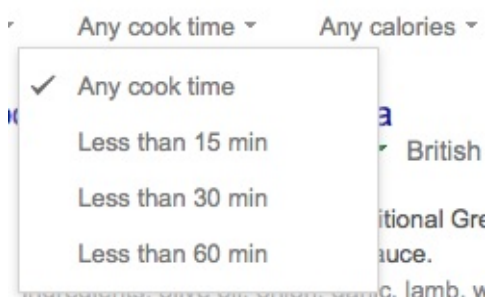
Solo tienes que hacer un par de clics para personalizar y filtrar los resultados de búsqueda para obtener recetas con los ingredientes, el tiempo de preparación o el aporte calórico perfectos para ti.

Ahora realizaremos una búsqueda de ejemplo. Si pulsamos en el link de la página anterior, puede ser que nos salga el mensaje: "La opción de búsqueda seleccionada no se encuentra disponible actualmente". Si ese es el caso, accederemos a la búsqueda de ejemplo desde [aquí](#) (el dominio debe ser .com y el idioma "en" para que podamos acceder a esta funcionalidad).

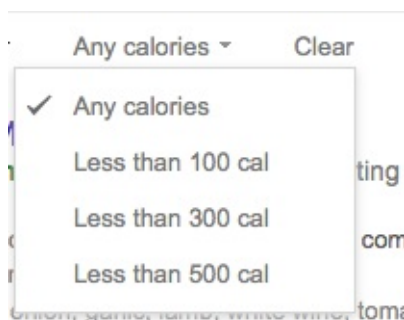
Buscamos el término "mousaka" y nos aparecen distintas recetas. Si pulsamos en "Ingredients" podemos indicar qué ingredientes queremos que utilice la receta y cuáles no, lo cual es muy útil para alérgicos, intolerantes o simplemente tenemos los ingredientes justos. Eliminamos por ejemplo los resultados que contengan vino tinto (Red wine) y yogurt (Yoghurt). Prestad atención al segundo enlace que vemos en la imagen porque lo usaremos en el apartado siguiente.



También podremos indicar el tiempo de elaboración que necesita el plato.



Incluso el número de calorías aproximado que contiene.



Esto es gracias a que el buscador de Google no está buscando ya sólo en el contenido de la página, sino en el microformato que alguien ha incluido en ésta. Veremos exactamente cómo es ese microformato en el apartado siguiente.

## Visualizar el microformato de una página

Para comprobar si una página web contiene un microformato y además visualizarlo con una estética amigable, podemos utilizar la herramienta "[Structured Data Testing Tool](#)" de Google. El funcionamiento es muy sencillo, a la derecha pegamos el código fuente de la página web o la URL y, después de analizarlo, nos aparecerá el microformato extraído a la derecha.

Testing Tool  2.192

<a href="#">Obtener la URL</a> <a href="#">Ejemplos</a> <span>VALIDAR</span>	Todavía no hay ningún resultado
<div><h2>1.</h2><h3>Introduce el código fuente</h3><p>Pégalo aquí O <a href="#">Obtener la URL</a></p></div>	<div><h2>2.</h2><h3>Valida los datos estructurados</h3><p>Pulsa el botón Validar y obtén los resultados aquí</p></div>

Pulsamos en "Obtener la URL" y ponemos la dirección "[http://www.bbc.co.uk/food/recipes/moussaka\\_6812](http://www.bbc.co.uk/food/recipes/moussaka_6812)", que es uno de los resultados que nos aparecieron antes. Pulsamos ahora en "Obtener y validar".

<input type="text" value="bc.co.uk/food/recipes/moussaka_6812"/>	<span>OBTENER Y VALIDAR</span>	<span>CANCELAR</span>
--	--------------------------------	-----------------------

Veremos como a la derecha nos aparece el microformato en una tabla fácilmente legible. Podemos ver como el formato se llama "hrecipe". En la tabla aparece el nombre, el resumen, un enlace hacia una foto... Y muchos más datos sobre la receta. En la imagen también se pueden ver dos etiquetas interesantes que indican que no contiene cacahuetes ni marisco respectivamente (muy útil para alérgicos).

▼ hrecipe (1)

Todo correcto ✓

hrecipe	
<b>fn:</b>	Moussaka
<b>summary:</b>	Indulge in a Greek-style moussaka, made with aubergines and lamb, packed with herbs and topped with two types of cheese.
<b>photo:</b>	<a href="http://ichef.bbci.co.uk/food/ic/food_16x9_448/recipes/moussaka_6812_16x9.jpg">http://ichef.bbci.co.uk/food/ic/food_16x9_448/recipes/moussaka_6812_16x9.jpg</a>
<b>yield:</b>	Serves 6
<b>tag:</b>	Nut-free
<b>tag:</b>	Shellfish-free
<b>preptime:</b>	PT1H
<b>instructions:</b>	Heat 2 tbsp olive oil in a large casserole dish. Add the onion and cook gently for 10 minutes or until the onion is soft, but not coloured. Brown half the lamb in a frying pan and add to the...

Más abajo podremos ver también cada uno de los ingredientes que componen la receta. Así que ya sabemos cómo la búsqueda anterior en Google podía discriminar entre los distintos ingredientes.

<b>author:</b>	Antony Worrall Thompson
<b>ingredient [ingredientinfo]:</b>	
<b>name:</b>	onion
<b>ingredient [ingredientinfo]:</b>	
<b>name:</b>	lamb mince
<b>ingredient [ingredientinfo]:</b>	
<b>name:</b>	garlic
<b>ingredient [ingredientinfo]:</b>	
<b>name:</b>	cinnamon
<b>ingredient [ingredientinfo]:</b>	
<b>name:</b>	allspice
<b>ingredient [ingredientinfo]:</b>	
<b>name:</b>	chopped tomatoes
<b>ingredient [ingredientinfo]:</b>	
<b>name:</b>	oregano
<b>ingredient [ingredientinfo]:</b>	
<b>value:</b>	2 bay leaves
<b>ingredient [ingredientinfo]:</b>	
<b>name:</b>	thyme
<b>ingredient [ingredientinfo]:</b>	
<b>name:</b>	white wine
<b>ingredient [ingredientinfo]:</b>	
<b>name:</b>	aubergines



Para que estos microdatos sean útiles, es fundamental que el conjunto de creadores de contenido web sean conscientes de la importancia de incluir los microformatos usando un vocabulario estándar (en este caso para las recetas el que aparece [aquí](#)) y rellenar todos los datos necesarios. Por ejemplo, si no incluimos el número de calorías aproximado, los buscadores que utilicen ese criterio como filtro eliminarán nuestro contenido de su resultado de búsqueda.



## Actualización [Noviembre 2015]

Actualmente, "Google Recipes" ha dejado de funcionar como tal. Sin embargo, aún podemos comprobar como Google hace uso de microformatos para refinar la búsqueda de blogs y patentes.

### Blogs

Para restringir una búsqueda a las entradas de los blogs, únicamente tenemos que añadir la cadena "&tbm=blg" al final de la dirección de búsqueda de Google. Por ejemplo:

<https://www.google.es/#q=microformatos&tbm=blg>



Web Maps Imágenes Vídeos Más ▾ Herramientas de búsqueda

Aproximadamente 4.260 resultados (0,19 segundos)

Las cookies nos permiten ofrecer nuestros servicios. Al utilizar nuestros servicios, aceptas el uso que hacemos de las cookies.

Más información [Entendido](#)

## Páginas principales de blogs sobre microformatos

### Microformats

[microformats.org/](https://microformats.org/) ▾ Traducir esta página

15 may. 2005 - The official microformats site. Contains the development wiki, mailing-list information and scripts to create the major ...

### Angal informática - Microformatos y el validador W3C

<https://www.angal.es/microformatos-validador-w3c/> ▾

25 ago. 2014 - Angal Informática actualiza su web con **microformatos** y realiza las modificaciones necesarias en el código para pasar el validador W3c.

### Welcome to the microformats wiki! · Microformats Wiki

[microformats.org/wiki/Main\\_Page](https://microformats.org/wiki/Main_Page) ▾ Traducir esta página

2 jul. 2015 - microformats are HTML for marking up people, organizations, events, locations, blog posts, products, reviews, resumes, recipes etc. Sites use microformats to publish a standard API that is consumed and used by search ...

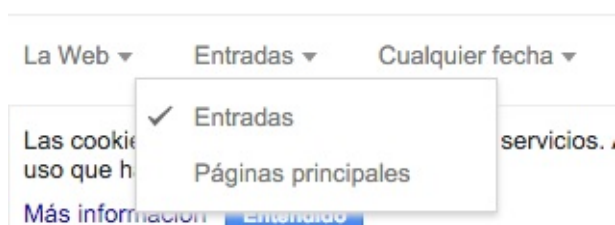
Más resultados de microformats.org

### ¿Utilizas los Microformatos en tu web? | Canal IP

[www.canalip.com/blog/¿utilizas-los-microformatos-en-tu-web/](http://www.canalip.com/blog/¿utilizas-los-microformatos-en-tu-web/) ▾

22 mar. 2011 - Los **microformatos**, según el W3C, son conjuntos de formatos de datos abiertos y simples, desarrollados sobre estándares ya existentes, ampliamente ad.

Podemos comprobar que si pulsamos en "Herramientas de búsqueda" aparece una nueva opción:



Así podemos restringir la búsqueda a las portadas de los blogs o a cualquier entrada de éstos.



## Patentes



Podemos poner otro ejemplo, esta vez relacionado con las patentes. Únicamente tendremos que añadir la cadena "&tbm=pts" si queremos que busque esa palabra entre las patentes registradas. Por ejemplo:

<https://www.google.es/#q=fregona&tbm=pts>

fregona



Web Imágenes Noticias Shopping Más ▾ Herramientas de búsqueda

Aproximadamente 84 resultados (0,39 segundos)

Las cookies nos permiten ofrecer nuestros servicios. Al utilizar nuestros servicios, aceptas el uso que hacemos de las cookies.

Más información Entendido

### Fregona - Wikipedia, la enciclopedia libre

<https://es.wikipedia.org/wiki/Fregona> ▾

La **fregona**, lampazo, trapeador, mapo, mopa, trapero, colete, suape, aljofifa, mocho, lava suelos, trapo de piso o mechudo es una herramienta para limpiar el ...

[Resultados de fregona »](#)

### Fregona con una esquina

image not available

[www.google.es/patents/WO2000071014A1?cl=es](https://www.google.es/patents/WO2000071014A1?cl=es)

Solicitada - Presentada el 11 May 2000 - Fecha de publicación: 30 Nov 2000 - Bernal Serradilla Ma Dolores - Bernal Serradilla Ma Dolores

**Fregona** con una esquina hace referencia a una **fregona** que, en la parte que se sujeta al mango y sostiene la mopa o pelos, tiene forma es ...

[Descripción general](#) - [Resultados relacionados](#) - [Foro de debate](#)

### Cabezal para mocho de fregona

image not available

[www.google.es/patents/WO1999029226A1?cl=es](https://www.google.es/patents/WO1999029226A1?cl=es)

Solicitada - Presentada el 4 Dic 1998 - Fecha de publicación: 17 Jun 1999 - [Escribano Gregorio Callejo](#) - [Escribano Gregorio Callejo](#)

Cabezal para mocho de **fregona**, que consiste en un cuerpo ...

[Descripción general](#) - [Resultados relacionados](#) - [Foro de debate](#)

Vemos que esta vez en "Herramientas de búsqueda" tenemos aún más opciones:

Cualquier oficina de patentes ▾    Cualquier estado ▾    Cualquier tipo de patente ▾

Por ejemplo, podemos elegir entre distintos tipos de patentes.

Cualquier oficina de patentes ▼

Cualquier estado ▼

Cualquier tipo de patente ▼

Ordenado

stros servicios. Al utilizar nuestros servicios, aceptas el

✓

Cualquier tipo de patente

Utilidad

Diseño

Planta

Publicación defensiva

Mejora adicional

Registro de invención legal

diclopedia libre

1a ▼

io, mopa, trapero, colete, suape, aljofifa, mocho,

es una herramienta para limpiar el ...

## Ejemplos de vocabularios útiles

Los vocabularios se utilizan para estandarizar las referencias a ciertos elementos. Así, si por ejemplo todo el mundo referencia dentro de un HTML el nombre de una persona de la misma forma con una palabra clave, los buscadores y motores podrán programarse para clasificar ese nombre como el de una persona. Proporcionando así semántica a los datos. En este capítulo veremos algunos de los vocabularios más utilizados referenciados por schema.org y la W3C.

### schema.org

[Schema.org](https://schema.org) proporciona una colección de "esquemas" (o vocabularios) que los creadores de contenido web pueden usar en las marcas HTML y que van a ser reconocidos por la mayoría de los buscadores (por ejemplo Bing, Google, Yahoo! y Yandex que utilizan los vocabularios de este sitio).

El sitio web es muy simple. En la pestaña de "Documentation" tenemos una introducción que nos explica cómo usar esos esquemas y en la pestaña "Schemas" podemos encontrarlos todos. Vamos a ver como ejemplo los esquemas "Person" (para personas físicas), "Movie" (para películas) y "Restaurant" (para restaurantes).

### Person

**Thing > Person**

A person (alive, dead, undead, or fictional).

Property	Expected Type	Description
<b>Properties from <u>Person</u></b>		
<u>additionalName</u>	Text	An additional name for a Person, can be used for a middle name.
<u>address</u>	PostalAddress	Physical address of the item.
<u>affiliation</u>	Organization	An organization that this person is affiliated with. For example, a school/university, a club, or a team.
<u>alumniOf</u>	EducationalOrganization	An educational organizations that the person is an alumni of. Inverse property: <u>alumni</u> .
<u>award</u>	Text	An award won by this person or for this creative work. Supersedes <u>awards</u> .
<u>birthDate</u>	Date	Date of birth.
<u>birthPlace</u>	Place	The place where the person was born.
<u>brand</u>	Brand or Organization	The brand(s) associated with a product or service, or the brand(s) maintained by an organization or business person.
<u>children</u>	Person	A child of the person.
<u>colleague</u>	Person	A colleague of the person. Supersedes <u>colleagues</u> .
<u>contactPoint</u>	ContactPoint	A contact point for a person or organization. Supersedes <u>contactPoints</u> .
<u>deathDate</u>	Date	Date of death.
<u>deathPlace</u>	Place	The place where the person died.
<u>duns</u>	Text	The Dun & Bradstreet DUNS number for identifying an organization or business person.
<u>email</u>	Text	Email address.
<u>familyName</u>	Text	Family name. In the U.S., the last name of an Person. This can be used along with givenName instead of the name property.

Como podemos ver en la imagen, Person deriva de Thing (cosa). Se utiliza para personas vivas, fallecidas o personajes de ficción. Tenemos una tabla con tres columnas. En la primera, "Property", tenemos el nombre de la propiedad (el que debemos poner en la marca HTML). En la segunda, "Expected Type", nos indica el tipo de dato que debemos asignar a esa propiedad (ya sea un texto, una fecha, un lugar...). Por último, en la columna "Description" nos muestra una corta descripción sobre a qué se refiere la propiedad.

Las propiedades están ordenadas por orden alfabético. Podemos ver en la captura por ejemplo como están "address" (para la dirección física de la persona), "birthDate" (para la fecha de nacimiento) o "children" (que se relaciona con otras personas e indica que son hijos o hijas de ésta).

Más abajo, en la misma tabla, podemos ver las propiedades que hereda o comparte con "Thing". Así, "name" (nombre) o "url" son propiedades que tienen el mismo nombre para todas las cosas.

<b>Properties from <u>Thing</u></b>		
<u>additionalType</u>	URL	An additional type for the item, typically used for adding more specific types from external vocabularies in microdata syntax. This is a relationship between something and a class that the thing is in. In RDFa syntax, it is better to use the native RDFa syntax – the 'typeof' attribute – for multiple types. Schema.org tools may have only weaker understanding of extra types, in particular those defined externally.
<u>alternateName</u>	Text	An alias for the item.
<u>description</u>	Text	A short description of the item.
<u>image</u>	URL or ImageObject	An image of the item. This can be a <u>URL</u> or a fully described <u>ImageObject</u> .
<u>name</u>	Text	The name of the item.
<u>potentialAction</u>	Action	Indicates a potential Action, which describes an idealized action in which this thing would play an 'object' role.
<u>sameAs</u>	URL	URL of a reference Web page that unambiguously indicates the item's identity. E.g. the URL of the item's Wikipedia page, Freebase page, or official website.
<u>url</u>	URL	URL of the item.

Una vez acabada esta tabla, tenemos otra distinta. Ésta no indica las propiedades de "Person", sino las propiedades de otros esquemas que tienen como tipo de datos la referencia a una persona. Así, tenemos el nombre de la propiedad, en qué esquema aparece y una descripción. Por ejemplo, podemos ver "author" (autor) que aparece en "CreativeWork" (trabajo creativo).

Property	On Types	Description
<a href="#">accountablePerson</a>	<a href="#">CreativeWork</a>	Specifies the Person that is legally accountable for the CreativeWork.
<a href="#">acquiredFrom</a>	<a href="#">OwnershipInfo</a>	The organization or person from which the product was acquired.
<a href="#">actor</a>	<a href="#">VideoGameSeries</a> or <a href="#">RadioSeries</a> or <a href="#">Episode</a> or <a href="#">TVSeries</a> or <a href="#">MovieSeries</a> or <a href="#">VideoObject</a> or <a href="#">Movie</a> or <a href="#">Clip</a> or <a href="#">VideoGame</a>	An actor, e.g. in tv, radio, movie, video games etc. Actors can be associated with individual items or with a series, episode, clip. Supersedes <a href="#">actors</a> .
<a href="#">agent</a>	<a href="#">Action</a>	The direct performer or driver of the action (animate or inanimate). e.g. *John* wrote a book.
<a href="#">alumni</a>	<a href="#">EducationalOrganization</a>	Alumni of educational organization. inverse property: <a href="#">alumniOf</a> .
<a href="#">athlete</a>	<a href="#">SportsTeam</a>	A person that acts as performing member of a sports team; a player as opposed to a coach.
<a href="#">attendee</a>	<a href="#">Event</a>	A person or organization attending the event. Supersedes <a href="#">attendees</a> .
<a href="#">author</a>	<a href="#">CreativeWork</a>	The author of this content. Please note that author is special in that HTML 5 provides a special mechanism for indicating authorship via the rel tag. That is equivalent to this and may be used interchangeably.
<a href="#">awayTeam</a>	<a href="#">SportsEvent</a>	The away team in a sports event.
<a href="#">borrower</a>	<a href="#">LendAction</a>	A sub property of participant. The person that borrows the object being lent.

Por último, si nos vamos al final de la página, veremos varios ejemplos de aplicación. Podremos ver cada ejemplo únicamente en HTML, etiquetado con microdatos, en RDFa y en formato JSON-LD. Lo cual resulta extremadamente útil, porque se puede comprobar de un vistazo cómo se aplica este esquema en la práctica.

Without Markup
Microdata
RDFa
JSON-LD

```

Jane Doe

Professor
20341 Whitworth Institute
405 Whitworth
Seattle WA 98052
(425) 123-4567
<a href="mailto:jane-doe@xyz.edu">jane-doe@illinois.edu</a>
Jane's home page:
<a href="http://www.janedoe.com">janedoe.com</a>
Graduate students:
<a href="http://www.xyz.edu/students/alicejones.html">Alice Jones</a>
<a href="http://www.xyz.edu/students/bobsmith.html">Bob Smith</a>

```

Por ejemplo en la imagen de arriba tenemos en HTML la ficha de datos de una profesora, con su correo electrónico, su web y dos estudiantes suyos graduados. En la siguiente imagen veremos cómo se podría conseguir aplicar el esquema "Person" a estos datos para que puedan ser analizados semánticamente por un buscador.



Without Markup   **Microdata**   RDFa   JSON-LD

```

<div itemscope itemtype="http://schema.org/Person">
  <span itemprop="name">Jane Doe</span>
  
  <span itemprop="jobTitle">Professor</span>
  <div itemprop="address" itemscope itemtype="http://schema.org/PostalAddress">
    <span itemprop="streetAddress">
      20341 Whitworth Institute
      405 N. Whitworth
    </span>
    <span itemprop="addressLocality">Seattle</span>,
    <span itemprop="addressRegion">WA</span>
    <span itemprop="postalCode">98052</span>
  </div>
  <span itemprop="telephone">(425) 123-4567</span>
  <a href="mailto:jane-doe@xyz.edu" itemprop="email">
    jane-doe@xyz.edu</a>
  Jane's home page:
  <a href="http://www.janedoe.com" itemprop="url">janedoe.com</a>
  Graduate students:
  <a href="http://www.xyz.edu/students/alicejones.html" itemprop="colleague">
    Alice Jones</a>
  <a href="http://www.xyz.edu/students/bobsmith.html" itemprop="colleague">
    Bob Smith</a>
</div>

```

La aplicación del esquema es bastante sencilla. Únicamente tenemos que definir al principio con "itemscope itemtype" el esquema al que nos estamos refiriendo. Luego, solamente tenemos que poner "itemprop=Nombre\_Propiedad" en la etiqueta donde está contenido el dato que queremos identificar como propiedad. Tenemos más información en la documentación de schema.org.

## Movie

### Thing > CreativeWork > Movie

A movie.

Property	Expected Type	Description
<b>Properties from <u>Movie</u></b>		
<u>actor</u>	<u>Person</u>	An actor, e.g. in tv, radio, movie, video games etc. Actors can be associated with individual items or with a series, episode, clip. Supersedes <u>actors</u> .
<u>director</u>	<u>Person</u>	A director of e.g. tv, radio, movie, video games etc. content. Directors can be associated with individual items or with a series, episode, clip. Supersedes <u>directors</u> .
<u>duration</u>	<u>Duration</u>	The duration of the item (movie, audio recording, event, etc.) in <u>ISO 8601 date format</u> .
<u>musicBy</u>	<u>Person</u> or <u>MusicGroup</u>	The composer of the soundtrack.
<u>productionCompany</u>	<u>Organization</u>	The production company or studio responsible for the item e.g. series, video game, episode etc.
<u>trailer</u>	<u>VideoObject</u>	The trailer of a movie or tv/radio series, season, episode, etc.

Este esquema se utiliza para las películas. Vemos que hereda las propiedades de "CreativeWork", que a la vez comparte las de "Thing". Tiene muy pocas propiedades exclusivas de su tipo, ya que la mayoría son compartidas. Tenemos "actor", "director" y "musicby" para señalar a las personas que han participado en la película. "productionCompany", que indica la compañía que la ha producido. También tenemos "trailer" que como vemos es de tipo "VideoObject", es decir, enlazará a un objeto tipo vídeo que contendrá el tráiler de la película. Por último, vamos a señalar "duration", que indica la duración de la película. La peculiaridad de esta propiedad es que nos indica que debe

expresarse en formato "ISO 8601 date format". Éste es un estándar internacional para fechas y tiempos. Si pulsamos en el [enlace](#) nos lleva a una página en Wikipedia donde nos enseña a usarlo.

### Durations [\[edit\]](#)

Durations are a component of time intervals and define the amount of intervening time in a time interval. They should only be used as part of a time interval as prescribed by the standard. Time intervals are discussed in the next section.

Durations are represented by the format `P[n]Y[n]M[n]DT[n]H[n]M[n]S` or `P[n]W` as shown to the right. In these representations, the `[n]` is replaced by the value for each of the date and time elements that follow the `[n]`. Leading zeros are not required, but the maximum number of digits for each element should be agreed to by the communicating parties. The capital letters *P*, *Y*, *M*, *W*, *D*, *T*, *H*, *M*, and *S* are designators for each of the date and time elements and are not replaced.

`PnYnMnDTnHnMnS`

`PnW`

`P<date>T<time>`

- *P* is the duration designator (historically called "period") placed at the start of the duration representation.
- *Y* is the year designator that follows the value for the number of years.
- *M* is the month designator that follows the value for the number of months.
- *W* is the week designator that follows the value for the number of weeks.
- *D* is the day designator that follows the value for the number of days.
- *T* is the time designator that precedes the time components of the representation.
- *H* is the hour designator that follows the value for the number of hours.
- *M* is the minute designator that follows the value for the number of minutes.
- *S* is the second designator that follows the value for the number of seconds.

For example, "P3Y6M4DT12H30M5S" represents a duration of "three years, six months, four days, twelve hours, thirty minutes, and five seconds".

Abajo de la página del esquema volvemos a encontrar algunos ejemplos de aplicación.

## Restaurant

### Thing > Place > LocalBusiness > FoodEstablishment > Restaurant

A restaurant.

Property	Expected Type	Description
<b>Properties from FoodEstablishment</b>		
<a href="#">acceptsReservations</a>	Text or Boolean or URL	Indicates whether a FoodEstablishment accepts reservations. Values can be Boolean, an URL at which reservations can be made or (for backwards compatibility) the strings <code>yes</code> or <code>no</code> .
<a href="#">menu</a>	Text or URL	Either the actual menu or a URL of the menu.
<a href="#">servesCuisine</a>	Text	The cuisine of the restaurant.
<b>Properties from LocalBusiness</b>		
<a href="#">branchOf</a>	Organization	The larger organization that this local business is a branch of, if any.
<a href="#">currenciesAccepted</a>	Text	The currency accepted (in <a href="#">ISO 4217 currency format</a> ).
<a href="#">openingHours</a>	Duration	<p>The opening hours for a business. Opening hours can be specified as a weekly time range, starting with days, then times per day. Multiple days can be listed with commas ',' separating each day. Day or time ranges are specified using a hyphen '-'.</p> <ul style="list-style-type: none"> <li>- Days are specified using the following two-letter combinations: Mo, Tu, We, Th, Fr, Sa, Su.</li> <li>- Times are specified using 24:00 time. For example, 3pm is specified as 15:00.</li> <li>- Here is an example: <code>&lt;time itemprop="openingHours" datetime="Tu,Th 16:00-20:00"&gt;Tuesdays and Thursdays 4-8pm&lt;/time&gt;</code>.</li> <li>- If a business is open 7 days a week, then it can be specified as <code>&lt;time itemprop="openingHours" datetime="Mo-Su"&gt;Monday through Sunday, all day&lt;/time&gt;</code>.</li> </ul>
<a href="#">paymentAccepted</a>	Text	Cash, credit card, etc.
<a href="#">priceRange</a>	Text	The price range of the business, for example <code>\$\$\$</code> .

El esquema "Restaurant" tiene una característica que no habíamos visto antes, y es que no tiene ninguna propiedad propia. Hereda las propiedades de "FoodEstablishment" (Establecimiento de comida), que a su vez hereda de "LocalBusiness" (Comercio local), "Place" (Lugar) y "Thing" (Cosa). Simplemente si todas las propiedades pensadas para un restaurante se pueden aplicar también a cualquier establecimiento de comida, no hay

necesidad de repetirlas. Se crea el esquema para tener una referencia específica para los restaurantes, pero sin propiedades inherentes. Queda abierta la posibilidad de que en el futuro se agregue alguna.

También podemos observar como la propiedad "acceptsReservations" (acepta reservas) puede indicarse como texto, booleano o URL. Si se pone como texto, se pueden indicar las peculiaridades de la reserva. Un booleano es un dato con dos valores, en este caso, sí o no. Por último, podemos indicar directamente la URL donde se realizan las reservas.

El resto de información que nos muestra, es similar al visto en los dos ejemplos anteriores.

## W3C

El [W3C](#) (World Wide Web Consortium) es una comunidad internacional en la que se trabaja para marcar los estándares de la Web. Tienen una [página](#) donde recomiendan lo que llaman "Buenas Ontologías", ya que están bien documentadas y son muy utilizadas. En este apartado analizaremos tres de ellas: Dublin Core, FOAF y SIOC.



### Dublin Core (DC)

## Dublin Core Metadata Initiative

[Dublin Core](#) es un vocabulario RDFS sencillo para describir metadatos genéricos. Suele utilizar los prefijos "dc:" y "dcter:". Está dividido en dos vocabularios, elementos DC y términos DC.

### Elementos DC

Los elementos DC contienen 15 propiedades. Las propiedades son contributor, coverage, creator, date, description, format, identifier, language, publisher, relation, rights, source, subject, title y type. En esta [página](#) se puede ver una descripción de cada una de ellas.



<b>Term Name:</b>	<b>creator</b>
<b>URI:</b>	<a href="http://purl.org/dc/elements/1.1/creator">http://purl.org/dc/elements/1.1/creator</a>
<b>Label:</b>	Creator
<b>Definition:</b>	An entity primarily responsible for making the resource.
<b>Comment:</b>	Examples of a Creator include a person, an organization, or a service. Typically, the name of a Creator should be used to indicate the entity.
<b>Type of Term:</b>	<u>Property</u>
<b>Version:</b>	<a href="http://dublincore.org/usage/terms/history/#creator-006">http://dublincore.org/usage/terms/history/#creator-006</a>
<b>Note:</b>	A second property with the same name as this property has been declared in the dcterms: namespace ( <a href="http://purl.org/dc/terms/">http://purl.org/dc/terms/</a> ). See the Introduction to the document "DCMI Metadata Terms" ( <a href="http://dublincore.org/documents/dcmi-terms/">http://dublincore.org/documents/dcmi-terms/</a> ) for an explanation.

En la imagen podemos ver la ficha de la propiedad "creator". Podemos ver en la definición que esta propiedad se refiere a la entidad primaria responsable de la creación de un recurso. Por ejemplo, el autor de un libro podríamos etiquetarlo como "dc:creator".

## Términos DC

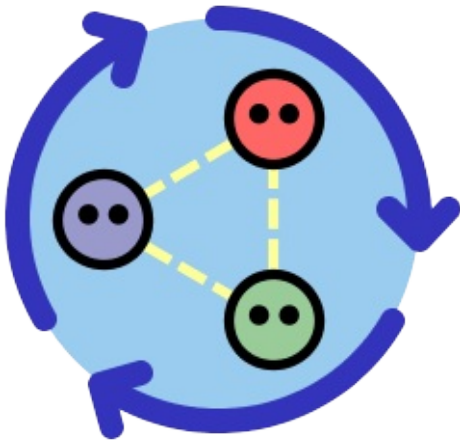
Los términos DC tienen 22 clases y 55 propiedades. En esta [página](#) podemos encontrar la descripción de cada una de las propiedades.

<b>Term Name:</b>	<b>language</b>
<b>URI:</b>	<a href="http://purl.org/dc/terms/language">http://purl.org/dc/terms/language</a>
<b>Label:</b>	Language
<b>Definition:</b>	A language of the resource.
<b>Comment:</b>	Recommended best practice is to use a controlled vocabulary such as RFC 4646 [RFC4646].
<b>References:</b>	[RFC4646] <a href="http://www.ietf.org/rfc/rfc4646.txt">http://www.ietf.org/rfc/rfc4646.txt</a>
<b>Type of Term:</b>	<u>Property</u>
<b>Refines:</b>	<a href="http://purl.org/dc/elements/1.1/language">http://purl.org/dc/elements/1.1/language</a>
<b>Has Range:</b>	<a href="http://purl.org/dc/terms/LinguisticSystem">http://purl.org/dc/terms/LinguisticSystem</a>
<b>Version:</b>	<a href="http://dublincore.org/usage/terms/history/#languageT-001">http://dublincore.org/usage/terms/history/#languageT-001</a>

En la imagen podemos ver la ficha de la propiedad "language". Como su definición nos explica, indica el idioma en el que se encuentra el recurso. Si, por ejemplo, el idioma de un recurso fuera el castellano, podríamos ponerlo en HTML así:

```
<span dcterms:language>Castellano</span>
```

## Friend Of A Friend (FOAF)



**Friend Of A Friend** es una ontología utilizada para describir personas y relaciones sociales en la Web. Está enfocada básicamente en la presencia de las personas en el mundo virtual, con muchas propiedades relaciones con su identidad y actividad online. No tiene nada sobre relaciones familiares o direcciones físicas. Utiliza el prefijo "foaf:".

Podemos utilizar la herramienta **FOAF-a-Matic** para crear un archivo que use esta ontología simplemente rellenando un formulario con los datos. La herramienta generará la información estructurada y etiquetada semánticamente de forma automática.

FOAF utiliza 19 clases, 44 propiedades de objetos y 27 propiedades de tipos de datos. Podemos ver una descripción de todo esto en su [especificación](#). En la siguiente tabla tenemos la relación de clases y propiedades.

```
Classes: | Agent | Document | Group | Image | LabelProperty | OnlineAccount | OnlineChatAccount | OnlineEcommerceAccount | OnlineGamingAccount | Organization | Person | PersonalProfileDocument | Project |

Properties: | account | accountName | accountServiceHomepage | age | aimChatID | based\_near | birthday | currentProject | depiction | depicts | dnaChecksum | familyName | family\_name | firstName | focus | fundedBy | geekcode | gender | givenName | givenname | holdsAccount | homepage | icqChatID | img | interest | isPrimaryTopicOf | jabberID | knows | lastName | logo | made | maker | mbox | mbox\_sha1sum | member | membershipClass | msnChatID | myersBriggs | name | nick | openid | page | pastProject | phone | plan | primaryTopic | publications | schoolHomepage | sha1 | skypeID | status | surname | theme | thumbnail | tipjar | title | topic | topic\_interest | weblog | workInfoHomepage | workplaceHomepage | yahooChatID |
```

En esta imagen vemos la ficha de la clase "foaf:Person". Nos muestra las propiedades que tiene, la relación con la que se usa, de qué es subclase y con qué clases es incompatible ("Disjoint" quiere decir por ejemplo que un mismo individuo no puede ser "Person" y "Project" al mismo tiempo). Además, nos muestra una descripción sobre a qué se refiere con esta clase. Podemos leer que representa a las personas, lo mismo si están vivas, fallecidas, son reales o imaginarias.

## Class: foaf:Person

*Person* - A person.

**Status:** stable

**Properties include:** [plan](#) [surname](#) [geekcode](#) [pastProject](#) [lastName](#) [family\\_name](#) [publications](#) [currentProject](#) [familyName](#) [firstName](#) [workInfoHomepage](#) [myersBriggs](#) [schoolHomepage](#) [img](#) [workplaceHomepage](#) [knows](#)

**Used with:** [knows](#)

**Subclass Of** [Agent](#) [Spatial Thing](#)

**Disjoint With:** [Project](#) [Organization](#)

The [Person](#) class represents people. Something is a [Person](#) if it is a person. We don't nitpic about whether they're alive, dead, real, or imaginary. The [Person](#) class is a sub-class of the [Agent](#) class, since all people are considered 'agents' in FOAF.

[#] [\[back to top\]](#)

En la siguiente imagen podemos ver la ficha de una de las propiedades de "foaf:Person", que es "foaf:workplaceHomepage". Esta propiedad se refiere a la página web del lugar de trabajo de la persona en cuestión. Nos aparece el dominio y el rango. El dominio se refiere desde dónde tiene que partir una propiedad (en este caso, de una persona). Por el contrario, el rango se refiere a dónde tiene que apuntar una propiedad (en este caso, a un documento). Incluso nos remarca un ejemplo de uso donde podemos comprobar como se aplica esta propiedad a una persona de forma práctica.

## Property: foaf:workplaceHomepage

*workplace homepage* - A workplace homepage of some person; the homepage of an organization they work for.

**Status:** testing

**Domain:** having this property implies being a [Person](#)

**Range:** every value of this property is a [Document](#)

The [workplaceHomepage](#) of a [Person](#) is a [Document](#) that is the [homepage](#) of a [Organization](#) that they work for.

By directly relating people to the homepages of their workplace, we have a simple convention that takes advantage of a set of widely known identifiers, while taking care not to confuse the things those identifiers identify (ie. organizational homepages) with the actual organizations those homepages describe.

For example, Dan Brickley works at W3C. Dan is a [Person](#) with a [homepage](#) of <http://danbri.org/>; W3C is a [Organization](#) with a [homepage](#) of <http://www.w3.org/>. This allows us to say that Dan has a [workplaceHomepage](#) of <http://www.w3.org/>.

```
<foaf:Person>
  <foaf:name>Dan Brickley</foaf:name>
  <foaf:workplaceHomepage rdf:resource="http://www.w3.org/" />
</foaf:Person>
```

Note that several other FOAF properties work this way; [schoolHomepage](#) is the most similar. In general, FOAF often indirectly identifies things via Web page identifiers where possible, since these identifiers are widely used and known. FOAF does not currently have a term for the name of the relation (eg. "workplace") that holds between a [Person](#) and an [Organization](#) that they work for.

[#] [\[back to top\]](#)



## Socially Interconnected Online Communities (SIOC)



[Socially Interconnected Online Communities](#) es una ontología utilizada para describir comunidades online como foros, blogs, listas de correo, wikis, etc. Complementa a FOAF haciendo hincapié en la descripción de los productos de esas comunidades (posts, respuestas, hilos y demás). Utiliza el prefijo "sioc:".

Wordpress y Drupal tienen un módulo que soporta esta ontología. Puedes ver un listado completo de herramientas que la utilizan [aquí](#).

SIOC utiliza 17 clases, 61 propiedades de objetos y 25 propiedades de tipos de datos. Podemos ver una descripción de todo esto en su [especificación](#). En la siguiente tabla tenemos la relación de clases y propiedades.

Classes:	<a href="#">Community</a>   <a href="#">Container</a>   <a href="#">Forum</a>   <a href="#">Item</a>   <a href="#">Post</a>   <a href="#">Role</a>   <a href="#">Site</a>   <a href="#">Space</a>   <a href="#">Thread</a>   <a href="#">UserAccount</a>   <a href="#">Usergroup</a>
Properties:	<a href="#">about</a>   <a href="#">account_of</a>   <a href="#">addressed_to</a>   <a href="#">administrator_of</a>   <a href="#">attachment</a>   <a href="#">avatar</a>   <a href="#">container_of</a>   <a href="#">content</a>   <a href="#">creator_of</a>   <a href="#">earlier_version</a>   <a href="#">email</a>   <a href="#">email_sha1</a>   <a href="#">embeds_knowledge</a>   <a href="#">feed</a>   <a href="#">follows</a>   <a href="#">function_of</a>   <a href="#">has_administrator</a>   <a href="#">has_container</a>   <a href="#">has_creator</a>   <a href="#">has_discussion</a>   <a href="#">has_function</a>   <a href="#">has_host</a>   <a href="#">has_member</a>   <a href="#">has_moderator</a>   <a href="#">has_modifier</a>   <a href="#">has_owner</a>   <a href="#">has_parent</a>   <a href="#">has_reply</a>   <a href="#">has_scope</a>   <a href="#">has_space</a>   <a href="#">has_subscriber</a>   <a href="#">has_usergroup</a>   <a href="#">host_of</a>   <a href="#">id</a>   <a href="#">ip_address</a>   <a href="#">last_activity_date</a>   <a href="#">last_item_date</a>   <a href="#">last_reply_date</a>   <a href="#">later_version</a>   <a href="#">latest_version</a>   <a href="#">link</a>   <a href="#">links_to</a>   <a href="#">member_of</a>   <a href="#">moderator_of</a>   <a href="#">modifier_of</a>   <a href="#">name</a>   <a href="#">next_by_date</a>   <a href="#">next_version</a>   <a href="#">note</a>   <a href="#">num_authors</a>   <a href="#">num_items</a>   <a href="#">num_replies</a>   <a href="#">num_threads</a>   <a href="#">num_views</a>   <a href="#">owner_of</a>   <a href="#">parent_of</a>   <a href="#">previous_by_date</a>   <a href="#">previous_version</a>   <a href="#">related_to</a>   <a href="#">reply_of</a>   <a href="#">scope_of</a>   <a href="#">sibling</a>   <a href="#">space_of</a>   <a href="#">subscriber_of</a>   <a href="#">topic</a>   <a href="#">usergroup_of</a>

A continuación en la siguiente imagen podemos ver la descripción de la clase "Forum". Es la clase propia de los foros. Nos indica que es subclase de "sioc:Container" (Contenedor), en qué rangos y dominios está incluido (explicados en el apartado anterior). Además incluye una extensa descripción sobre cómo son tratados los foros.

## Class: sioc:Forum

*Forum* - A discussion area on which Posts or entries are made.

**sub-class-of:** [sioc:Container](#)

**in-range-of:** [sioc:host\\_of](#) [sioc:moderator\\_of](#)

**in-domain-of:** [sioc:has\\_host](#) [sioc:has\\_moderator](#) [sioc:num\\_threads](#)

Forums can be thought of as channels or discussion area on which Posts are made. A Forum can be linked to the Site that hosts it. Forums will usually discuss a certain topic or set of related topics, or they may contain discussions entirely devoted to a certain community group or organisation. A Forum will have a moderator who can veto or edit posts before or after they appear in the Forum.

Forums may have a set of subscribed UserAccounts who are notified when new Posts are made. The hierarchy of Forums can be defined in terms of parents and children, allowing the creation of structures conforming to topic categories as defined by the Site administrator. Examples of Forums include mailing lists, message boards, Usenet newsgroups and weblogs.

The [SIOC Types Ontology Module](#) defines some more specific subclasses of sioc:Forum.

También vamos a ver un ejemplo de cómo se describe una propiedad en la especificación. En este caso, "sioc:administrator\_of" (administrador de). La propiedad especifica que la cuenta de usuario es administradora de un sitio concreto. Por eso el dominio es "sioc:UserAccount" y el rango es "sioc:Site". Además nos indica que tiene una propiedad inversa, "sioc:has\_administrator".

## Property: sioc:administrator\_of

*administrator of* - A Site that the UserAccount is an administrator of.

**Inverse:** [sioc:has\\_administrator](#)

**OWL Type:** ObjectProperty

**Domain:** [sioc:UserAccount](#)

**Range:** [sioc:Site](#)

Por último os proponemos revisar el ejemplo expuesto en su especificación. En este caso, se describe la entrada de un blog.

```

<sioc:Post rdf:about="http://johnbreslin.com/blog/2006/09/07/creating-connections-between-discussion-clouds-with-sioc/">
  <dcterms:title>Creating connections between discussion clouds with SIOC</dcterms:title>
  <dcterms:created>2006-09-07T09:33:30Z</dcterms:created>
  <sioc:has_container rdf:resource="http://johnbreslin.com/blog/index.php?sioc_type=site#weblog"/>
  <sioc:has_creator>
    <sioc:UserAccount rdf:about="http://johnbreslin.com/blog/author/cloud/" rdfs:label="Cloud">
      <rdfs:seeAlso rdf:resource="http://johnbreslin.com/blog/index.php?sioc_type=user&sioc_id=1"/>
    </sioc:UserAccount>
  </sioc:has_creator>
  <sioc:content>SIOC provides a unified vocabulary for content and interaction description: a semantic layer that can co-exist with existing discussion platforms.</sioc:content>
  <sioc:topic rdfs:label="Semantic Web" rdf:resource="http://johnbreslin.com/blog/category/semantic-web/">
  <sioc:topic rdfs:label="Blogs" rdf:resource="http://johnbreslin.com/blog/category/blogs/">
  <sioc:has_reply>
    <sioc:Post rdf:about="http://johnbreslin.com/blog/2006/09/07/creating-connections-between-discussion-clouds-with-sioc/#comment-123928">
      <rdfs:seeAlso rdf:resource="http://johnbreslin.com/blog/index.php?sioc_type=comment&sioc_id=123928"/>
    </sioc:Post>
  </sioc:has_reply>
</sioc:Post>

```

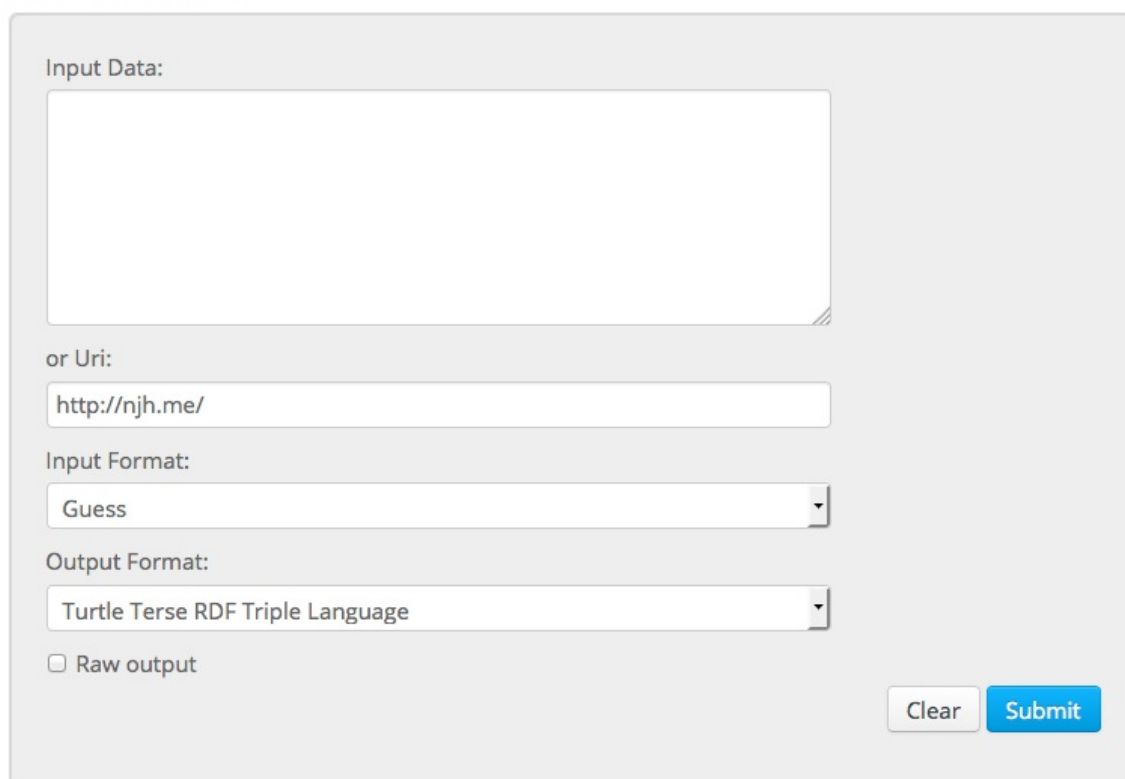


## Ejemplo de conversor de sintaxis RDF

[EasyRDF](#) es una librería en lenguaje PHP diseñada para hacer sencillo el consumo y la producción de RDF.

Nos centraremos en una de sus características que nos permite convertir entre distintos tipos de sintaxis RDF. Incluso podemos hacerlo directamente en la misma página web de la herramienta a la que accederemos [aquí](#).

### Converter



The screenshot shows the 'Converter' web interface. It has a light gray background. At the top, there's a label 'Input Data:' above a large, empty white text area. Below this, there's a label 'or Uri:' above a text input field containing 'http://njh.me/'. Underneath that is a label 'Input Format:' above a dropdown menu showing 'Guess'. Below the dropdown is a label 'Output Format:' above another dropdown menu showing 'Turtle Terse RDF Triple Language'. At the bottom left, there's a checkbox labeled 'Raw output' which is currently unchecked. At the bottom right, there are two buttons: a 'Clear' button and a 'Submit' button.

This converter is running version **0.9.0** of EasyRdf.

Usaremos como ejemplo el [archivo](#) generado en el [capítulo anterior](#). Este archivo lo generamos en formato Turtle, así que vamos a convertirlo a formato RDF/XML. Para ello copiamos el contenido del archivo y lo pegamos en "Input Data:". En "or URI" pondremos la URI general del archivo, en este caso "<http://ejemplo.org/>". En "Input Format:" pondremos "Turtle Terse RDF Triple Language" y en "Output Format:" pondremos "RDF/XML". Y pulsamos el botón "Submit".

# Converter

Input Data:

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix dbpedia-owl: <http://dbpedia.org/ontology/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
```

or Uri:

Input Format:

Output Format:

☐ Raw output

Nos aparecerá en la parte inferior el contenido del archivo convertido al formato RDF/XML.

## Output

Number of triples parsed: 59

```
<?xml version="1.0" encoding="utf-8" ?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:ns0="http://dbpedia.org/ontology/">

  <foaf:Person rdf:about="http://ejemplo.org/director/1">
    <rdf:type rdf:resource="http://dbpedia.org/ontology/MovieDirector"/>
    <foaf:name>Álex de la Iglesia</foaf:name>
    <owl:sameAs rdf:resource="http://es.dbpedia.org/resource/Alex_de_la_Iglesia"/>
  </ns0:director>

  <ns0:Film rdf:about="http://ejemplo.org/film/4">
    <foaf:name>El día de la bestia</foaf:name>
    <owl:sameAs rdf:resource="http://es.dbpedia.org/resource/El_dia_de_la_Bestia"/>
  </ns0:Film>
</ns0:director>

  <ns0:director>
    <ns0:Film rdf:about="http://ejemplo.org/film/8">
      <foaf:name>Balada triste de trompeta</foaf:name>
      <owl:sameAs rdf:resource="http://es.dbpedia.org/resource/Balada_triste_de_trompeta"/>
    </ns0:Film>
  </ns0:director>
```

Si marcamos la casilla "Raw output", en lugar de aparecernos abajo el resultado, nos devolverá directamente el archivo transformado a la nueva sintaxis.

Este fichero XML no parece tener ninguna información de estilo asociada. Se muestra debajo el árbol del documento.

```
- <rdf:RDF>
- <foaf:Person rdf:about="http://ejemplo.org/director/1">
  <rdf:type rdf:resource="http://dbpedia.org/ontology/MovieDirector"/>
  <foaf:name>Álex de la Iglesia</foaf:name>
  <owl:sameAs rdf:resource="http://es.dbpedia.org/resource/Alex_de_la_Iglesia"/>
- <ns0:director>
  - <ns0:Film rdf:about="http://ejemplo.org/film/4">
    <foaf:name>El día de la bestia</foaf:name>
    <owl:sameAs rdf:resource="http://es.dbpedia.org/resource/El_dia_de_la_Bestia"/>
  </ns0:Film>
</ns0:director>
- <ns0:director>
  - <ns0:Film rdf:about="http://ejemplo.org/film/8">
    <foaf:name>Balada triste de trompeta</foaf:name>
    <owl:sameAs rdf:resource="http://es.dbpedia.org/resource/Balada_triste_de_trompeta"/>
  </ns0:Film>
</ns0:director>
</foaf:Person>
- <ns0:Film rdf:about="http://ejemplo.org/film/1">
  <foaf:name>El otro lado de la cama</foaf:name>
  <owl:sameAs rdf:resource="http://es.dbpedia.org/resource/El_otro_lado_de_la_cama"/>
</ns0:Film>
- <foaf:Person rdf:about="http://ejemplo.org/director/3">
  - <ns0:director>
    - <ns0:Film rdf:about="http://ejemplo.org/film/7">
      <foaf:name>Ocho apellidos vascos</foaf:name>
      <owl:sameAs rdf:resource="http://es.dbpedia.org/resource/Ocho_apellidos_vascos"/>
    </ns0:Film>
  </ns0:director>
  <rdf:type rdf:resource="http://dbpedia.org/ontology/MovieDirector"/>
  <foaf:name>Emilio Martínez-Lázaro</foaf:name>
  <owl:sameAs rdf:resource="http://es.dbpedia.org/resource/Emilio_Martinez-Lazaro"/>
</foaf:Person>
```

# Ejemplo de parseo programático de RDF

En este capítulo veremos como realizar un parseo (transformar un texto de entrada en una estructura de datos apropiada para ser procesada) de un archivo RDF usando el lenguaje de programación Python, que es uno de los más sencillos de usar y aprender. Para ello, usaremos la librería RDFlib.

## 1. Instalación de RDFlib

### 1.1 setuptools

Primero necesitaremos instalar este [paquete](#). Para ello descargaremos el archivo [ez\\_setup.py](#) y lo ejecutaremos como script de Python. Para ello, escribiremos en una consola:

```
Python ruta_del_archivo/ez_setup.py
```

### 1.2 RDFlib

Para la [instalación](#) de esta librería, tendremos ahora únicamente que escribir lo siguiente en una consola:

```
easy_install rdflib
```

Hay que tener en cuenta que para alguno de los dos comandos pueden ser necesarios privilegios de administrador. Con esto ya tendremos todo instalado.

## 2. Parseo de un fichero

Utilizaremos en este caso el mismo archivo que en ejemplos anteriores, [Películas-y-directores.ttl](#). Esta vez haremos un script en Python que sea capaz de leer el fichero utilizando RDFlib. A continuación tenemos el código comentado que utilizaremos:

```
from rdflib import Graph

g = Graph()
g.parse("Películas-y-directores.ttl", format="turtle")

print(len(g))

import pprint
for stmt in g:
    pprint.pprint(stmt)
```

En la primera línea importamos la clase que almacena los grafos RDF. En la segunda línea declaramos uno nuevo y en la tercera parseamos el fichero para almacenarlo en ese grafo. Tendremos que indicarle el nombre (o ruta) y el formato en el que se encuentra. Podremos parsear con RDFlib los siguientes formatos de fichero:

Nombre	Clase
html	StructuredDataParser
hturtle	HTurtleParser
mdata	MicrodataParser
microdata	MicrodataParser
n3	N3Parser
nquads	NQuadsParser
nt	NTParser
rdfa	RDFaParser
rdfa1.0	RDFa10Parser
rdfa1.1	RDFaParser
trix	TriXParser
turtle	TurtleParser
xml	RDFXMLParser

Nosotros hemos utilizado "turtle", que es el formato del que partía nuestro fichero. Con "print(len(g))" imprimirá por pantalla el número de sentencias turtle que tiene el fichero (que ahora es un grafo de Python). Y en el bucle del final, imprimimos cada una de ellas. El resultado por pantalla será el siguiente:

```
59
(rdflib.term.URIRef(u'http://ejemplo.org/director/1'),
 rdflib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
 rdflib.term.URIRef(u'http://dbpedia.org/ontology/MovieDirector'))
```

```

(rdfLib.term.URIRef(u'http://ejemplo.org/director/4'),
 rdfLib.term.URIRef(u'http://www.w3.org/2002/07/owl#sameAs'),
 rdfLib.term.URIRef(u'http://es.dbpedia.org/resource/Juan_Antonio_Bayona'))
(rdfLib.term.URIRef(u'http://ejemplo.org/film/10'),
 rdfLib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
 rdfLib.term.URIRef(u'http://dbpedia.org/ontology/Film'))
(rdfLib.term.URIRef(u'http://ejemplo.org/film/8'),
 rdfLib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
 rdfLib.term.URIRef(u'http://dbpedia.org/ontology/Film'))
(rdfLib.term.URIRef(u'http://ejemplo.org/film/3'),
 rdfLib.term.URIRef(u'http://xmlns.com/foaf/0.1/name'),
 rdfLib.term.Literal(u'La voz dormida'))
(rdfLib.term.URIRef(u'http://ejemplo.org/film/2'),
 rdfLib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
 rdfLib.term.URIRef(u'http://dbpedia.org/ontology/Film'))
(rdfLib.term.URIRef(u'http://ejemplo.org/director/1'),
 rdfLib.term.URIRef(u'http://www.w3.org/2002/07/owl#sameAs'),
 rdfLib.term.URIRef(u'http://es.dbpedia.org/resource/Alex_de_la_Iglesia'))
(rdfLib.term.URIRef(u'http://ejemplo.org/director/2'),
 rdfLib.term.URIRef(u'http://dbpedia.org/ontology/director'),
 rdfLib.term.URIRef(u'http://ejemplo.org/film/3'))
(rdfLib.term.URIRef(u'http://ejemplo.org/director/3'),
 rdfLib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
 rdfLib.term.URIRef(u'http://dbpedia.org/ontology/MovieDirector'))
(rdfLib.term.URIRef(u'http://ejemplo.org/director/2'),
 rdfLib.term.URIRef(u'http://dbpedia.org/ontology/director'),
 rdfLib.term.URIRef(u'http://ejemplo.org/film/10'))
(rdfLib.term.URIRef(u'http://ejemplo.org/film/3'),
 rdfLib.term.URIRef(u'http://www.w3.org/2002/07/owl#sameAs'),
 rdfLib.term.URIRef(u'http://es.dbpedia.org/resource/La_voz_dormida'))
(rdfLib.term.URIRef(u'http://ejemplo.org/director/3'),
 rdfLib.term.URIRef(u'http://www.w3.org/2002/07/owl#sameAs'),
 rdfLib.term.URIRef(u'http://es.dbpedia.org/resource/Emilio_Martinez-Lazaro'))
(rdfLib.term.URIRef(u'http://ejemplo.org/director/4'),
 rdfLib.term.URIRef(u'http://dbpedia.org/ontology/director'),
 rdfLib.term.URIRef(u'http://ejemplo.org/film/9'))
(rdfLib.term.URIRef(u'http://ejemplo.org/film/2'),
 rdfLib.term.URIRef(u'http://xmlns.com/foaf/0.1/name'),
 rdfLib.term.Literal(u'Mientras duermes'))
(rdfLib.term.URIRef(u'http://ejemplo.org/film/5'),
 rdfLib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
 rdfLib.term.URIRef(u'http://dbpedia.org/ontology/Film'))
(rdfLib.term.URIRef(u'http://ejemplo.org/film/7'),
 rdfLib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
 rdfLib.term.URIRef(u'http://dbpedia.org/ontology/Film'))
(rdfLib.term.URIRef(u'http://ejemplo.org/director/3'),
 rdfLib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
 rdfLib.term.URIRef(u'http://xmlns.com/foaf/0.1/Person'))
(rdfLib.term.URIRef(u'http://ejemplo.org/film/4'),
 rdfLib.term.URIRef(u'http://xmlns.com/foaf/0.1/name'),
 rdfLib.term.Literal(u'El d\xeda de la bestia'))
(rdfLib.term.URIRef(u'http://ejemplo.org/film/8'),
 rdfLib.term.URIRef(u'http://www.w3.org/2002/07/owl#sameAs'),

```



```

    rdflib.term.URIRef(u'http://es.dbpedia.org/resource/Balada_triste_de_trompeta'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/4'),
 rdflib.term.URIRef(u'http://xmlns.com/foaf/0.1/name'),
 rdflib.term.Literal(u'Juan Antonio Bayona'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/4'),
 rdflib.term.URIRef(u'http://www.w3.org/2002/07/owl#sameAs'),
 rdflib.term.URIRef(u'http://es.dbpedia.org/resource/El_dia_de_la_Bestia'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/3'),
 rdflib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
 rdflib.term.URIRef(u'http://dbpedia.org/ontology/Film'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/2'),
 rdflib.term.URIRef(u'http://www.w3.org/2002/07/owl#sameAs'),
 rdflib.term.URIRef(u'http://es.dbpedia.org/resource/Mientras_duermes'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/10'),
 rdflib.term.URIRef(u'http://www.w3.org/2002/07/owl#sameAs'),
 rdflib.term.URIRef(u'http://es.dbpedia.org/resource/Solas_(pelicula)'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/2'),
 rdflib.term.URIRef(u'http://xmlns.com/foaf/0.1/name'),
 rdflib.term.Literal(u'Benito Zambrano'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/5'),
 rdflib.term.URIRef(u'http://www.w3.org/2002/07/owl#sameAs'),
 rdflib.term.URIRef(u'http://es.dbpedia.org/resource/Lo_imposible'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/1'),
 rdflib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
 rdflib.term.URIRef(u'http://xmlns.com/foaf/0.1/Person'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/7'),
 rdflib.term.URIRef(u'http://www.w3.org/2002/07/owl#sameAs'),
 rdflib.term.URIRef(u'http://es.dbpedia.org/resource/Ocho_apellidos_vascos'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/2'),
 rdflib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
 rdflib.term.URIRef(u'http://xmlns.com/foaf/0.1/Person'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/5'),
 rdflib.term.URIRef(u'http://xmlns.com/foaf/0.1/name'),
 rdflib.term.Literal(u'Lo imposible'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/5'),
 rdflib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
 rdflib.term.URIRef(u'http://xmlns.com/foaf/0.1/Person'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/3'),
 rdflib.term.URIRef(u'http://xmlns.com/foaf/0.1/name'),
 rdflib.term.Literal(u'Emilio Mart\xednez-L\xedzaro'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/5'),
 rdflib.term.URIRef(u'http://xmlns.com/foaf/0.1/name'),
 rdflib.term.Literal(u'Jaume Balaguer\xfa'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/4'),
 rdflib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
 rdflib.term.URIRef(u'http://xmlns.com/foaf/0.1/Person'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/1'),
 rdflib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
 rdflib.term.URIRef(u'http://dbpedia.org/ontology/Film'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/1'),
 rdflib.term.URIRef(u'http://www.w3.org/2002/07/owl#sameAs'),
 rdflib.term.URIRef(u'http://es.dbpedia.org/resource/El_otro_lado_de_la_cama'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/5'),

```

```

rdflib.term.URIRef(u'http://www.w3.org/2002/07/owl#sameAs'),
rdflib.term.URIRef(u'http://es.dbpedia.org/resource/Jaume_Balaguero'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/10'),
rdflib.term.URIRef(u'http://xmlns.com/foaf/0.1/name'),
rdflib.term.Literal(u'Solas'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/1'),
rdflib.term.URIRef(u'http://xmlns.com/foaf/0.1/name'),
rdflib.term.Literal(u'El otro lado de la cama'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/8'),
rdflib.term.URIRef(u'http://xmlns.com/foaf/0.1/name'),
rdflib.term.Literal(u'Balada triste de trompeta'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/5'),
rdflib.term.URIRef(u'http://dbpedia.org/ontology/director'),
rdflib.term.URIRef(u'http://ejemplo.org/film/6'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/7'),
rdflib.term.URIRef(u'http://xmlns.com/foaf/0.1/name'),
rdflib.term.Literal(u'Ocho apellidos vascos'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/1'),
rdflib.term.URIRef(u'http://dbpedia.org/ontology/director'),
rdflib.term.URIRef(u'http://ejemplo.org/film/4'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/6'),
rdflib.term.URIRef(u'http://xmlns.com/foaf/0.1/name'),
rdflib.term.Literal(u'Fr\xe1giles'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/9'),
rdflib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
rdflib.term.URIRef(u'http://dbpedia.org/ontology/Film'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/3'),
rdflib.term.URIRef(u'http://dbpedia.org/ontology/director'),
rdflib.term.URIRef(u'http://ejemplo.org/film/7'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/1'),
rdflib.term.URIRef(u'http://dbpedia.org/ontology/director'),
rdflib.term.URIRef(u'http://ejemplo.org/film/8'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/5'),
rdflib.term.URIRef(u'http://dbpedia.org/ontology/director'),
rdflib.term.URIRef(u'http://ejemplo.org/film/2'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/9'),
rdflib.term.URIRef(u'http://xmlns.com/foaf/0.1/name'),
rdflib.term.Literal(u'El orfanato'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/9'),
rdflib.term.URIRef(u'http://www.w3.org/2002/07/owl#sameAs'),
rdflib.term.URIRef(u'http://es.dbpedia.org/resource/El_orfanato'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/4'),
rdflib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
rdflib.term.URIRef(u'http://dbpedia.org/ontology/MovieDirector'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/4'),
rdflib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
rdflib.term.URIRef(u'http://dbpedia.org/ontology/Film'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/6'),
rdflib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
rdflib.term.URIRef(u'http://dbpedia.org/ontology/Film'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/2'),
rdflib.term.URIRef(u'http://www.w3.org/2002/07/owl#sameAs'),
rdflib.term.URIRef(u'http://es.dbpedia.org/resource/Benito_Zambrano'))

```

```
(rdflib.term.URIRef(u'http://ejemplo.org/director/1'),
 rdflib.term.URIRef(u'http://xmlns.com/foaf/0.1/name'),
 rdflib.term.Literal(u'\xc1lex de la Iglesia'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/5'),
 rdflib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
 rdflib.term.URIRef(u'http://dbpedia.org/ontology/MovieDirector'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/4'),
 rdflib.term.URIRef(u'http://dbpedia.org/ontology/director'),
 rdflib.term.URIRef(u'http://ejemplo.org/film/5'))
(rdflib.term.URIRef(u'http://ejemplo.org/film/6'),
 rdflib.term.URIRef(u'http://www.w3.org/2002/07/owl#sameAs'),
 rdflib.term.URIRef(u'http://es.dbpedia.org/resource/Fragiles_(pelicula)'))
(rdflib.term.URIRef(u'http://ejemplo.org/director/2'),
 rdflib.term.URIRef(u'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'),
 rdflib.term.URIRef(u'http://dbpedia.org/ontology/MovieDirector'))
```

También podemos realizar un parseo de un fichero que esté colgado en internet. Sería tan sencillo como escribir:

```
g.parse("http://bigasterisk.com/foaf.rdf")
```

Por lo tanto, sólo tenemos que poner la URL. En este caso no hemos especificado el formato, por defecto lo toma como "xml", que es el formato típico en el que están los archivos .rdf.

## Ejemplo de inferencia de relaciones

Los repositorios en los que almacenamos las ontologías OWL pueden ser RDF(S)-aware. Esto significa que es capaz de inferir nuevas relaciones entre entidades que no están definidas explícitamente a través de distintas propiedades como la transitividad. Aunque en la mayoría de los casos se usa un "razonador" (un motor que analiza las ontologías) aparte que detecta inconsistencias e infiere automáticamente relaciones y las coloca de forma explícita en el repositorio.

En este capítulo vamos a ver un ejemplo de cómo utilizar un razonador para inferir relaciones implícitas. Para ello, utilizaremos [Protégé](#). Éste es un editor de ontologías de código abierto y gratuito creado por la Universidad de Stanford. Por defecto, trae instalado el razonador [Hermit](#), aunque es posible añadirle otros.

## Descarga e instalación de Protégé

Nos dirigiremos a la página oficial de [descarga](#) y elegiremos la última versión disponible. A la hora de realizar este tutorial, es la versión 5.0.

La instalación es muy sencilla, aunque depende del sistema operativo. Podemos acudir a esta [página](#) si queremos instrucciones concretas o tenemos algún problema con la instalación. Para la versión 5.0 de Protégé es necesaria la versión 7 de [Java](#).

## Ejemplo sencillo de inferencia

Ilustraremos los pasos necesarios para ver un ejemplo de lo que puede hacer el razonador, pero no exploraremos a fondo las posibilidades y el funcionamiento de Protégé. Se puede acceder a una guía de uso sencilla del editor [aquí](#). También tenéis un completo tutorial que muestra paso por paso las posibilidades del editor con marco teórico incluido a través de un ejemplo en su página oficial [aquí](#). Y [aquí](#) si no funcionara el enlace anterior.

Al abrir por primera vez el editor, nos encontraremos una ventana parecida a la siguiente.

untitled-ontology-5 (<http://www.semanticweb.org/ontologies/2015/1/untitled-ontology-5>)

Search for entity

Active Ontology | Entities | Classes | Object Properties | Data Properties | Annotation Properties | Individuals | OWLViz | DL Query

**Ontology header:**

Ontology IRI <http://www.semanticweb.org/ontologies/2015/1/untitled-ontology-5>

Ontology Version IRI e.g. <http://www.semanticweb.org/ontologies/2015/1/untitled-ontology-5>

Annotations +

**Ontology metrics:**

Metrics	
Axiom	0
Logical axiom count	0
Class count	0
Object property count	0
Data property count	0
Individual count	0
DL expressivity	AL

**Class axioms**

SubClassOf axioms count	0
EquivalentClasses axioms count	0
DisjointClasses axioms count	0
GCI count	0
Hidden GCI Count	0

**Object property axioms**

Ontology imports | Ontology Prefixes | General class axioms

**Imported ontologies:**

Direct Imports +

Indirect Imports

To use the reasoner click Reasoner->Start reasoner ☒ Show Inferences

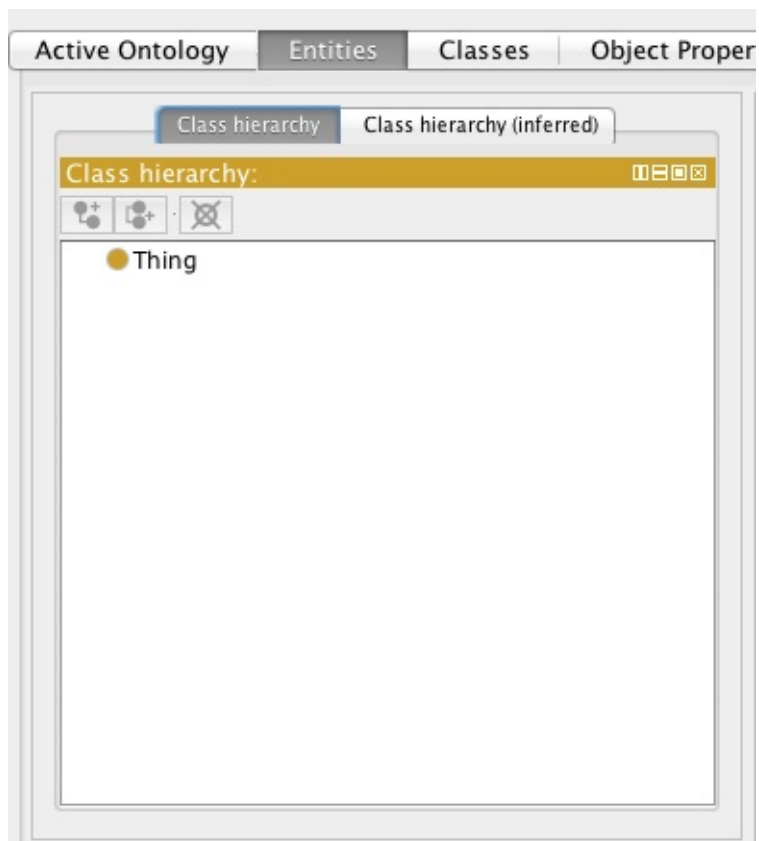
En "Ontology IRI" escribiremos la URI raíz que tendrá nuestra ontología. Por ejemplo, "<http://ejemplo.com/inferencia>".

**Ontology header:**

Ontology IRI <http://ejemplo.com/inferencia>

Ontology Version IRI e.g. <http://ejemplo.com/inferencia/1.0.0>

Ahora iremos a la pestaña "Entities" para definir las entidades que vamos a usar. Podemos fijarnos en que donde aparece la jerarquía de clases tenemos dos pestañas "Class Hierarchy" y "Class Hierarchy (Inferred)". En la pestaña de la izquierda es donde nosotros editaremos y en la de la derecha aparecerán las clases inferidas o marcadas como inconsistentes por el razonador.

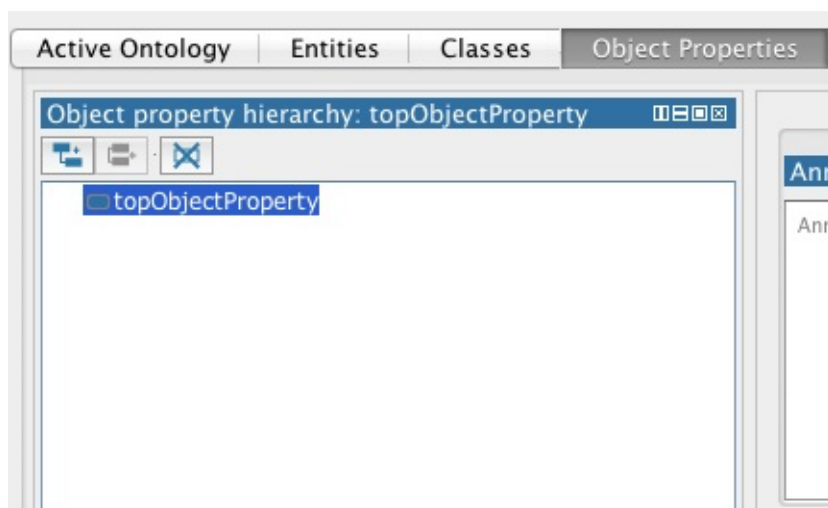


Únicamente utilizaremos la entidad "Persona". Para ello, haremos click en "Thing" y en el botón de arriba a la izquierda para crear una subclase de ésta.

The image shows a dialog box for creating a new entity. It has two input fields: 'Name' and 'IRI'. The 'Name' field contains the text 'Persona'. The 'IRI' field contains the text 'http://ejemplo.com/inferencia#Persona'. At the bottom right, there is a button labeled 'New entity options...'. The dialog box has a light gray background and a white border.

A continuación iremos a la pestaña "Object Properties" para definir las propiedades que puede tener un objeto. Todas las propiedades que creamos deben ser descendientes de "topObjectProperty", que es la que viene por defecto.



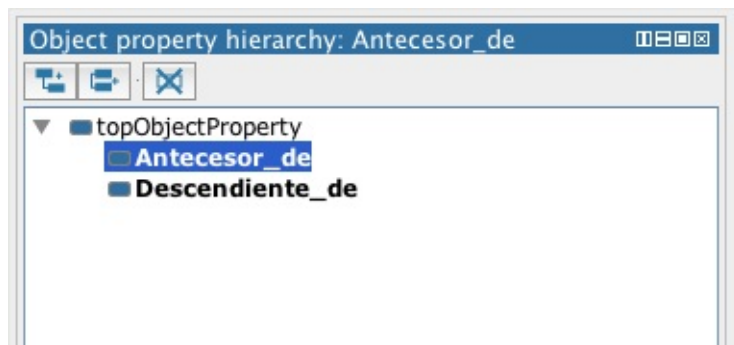


Hacemos click en ésta y en el botón de arriba a la izquierda para crear la propiedad "Descendiente\_de".

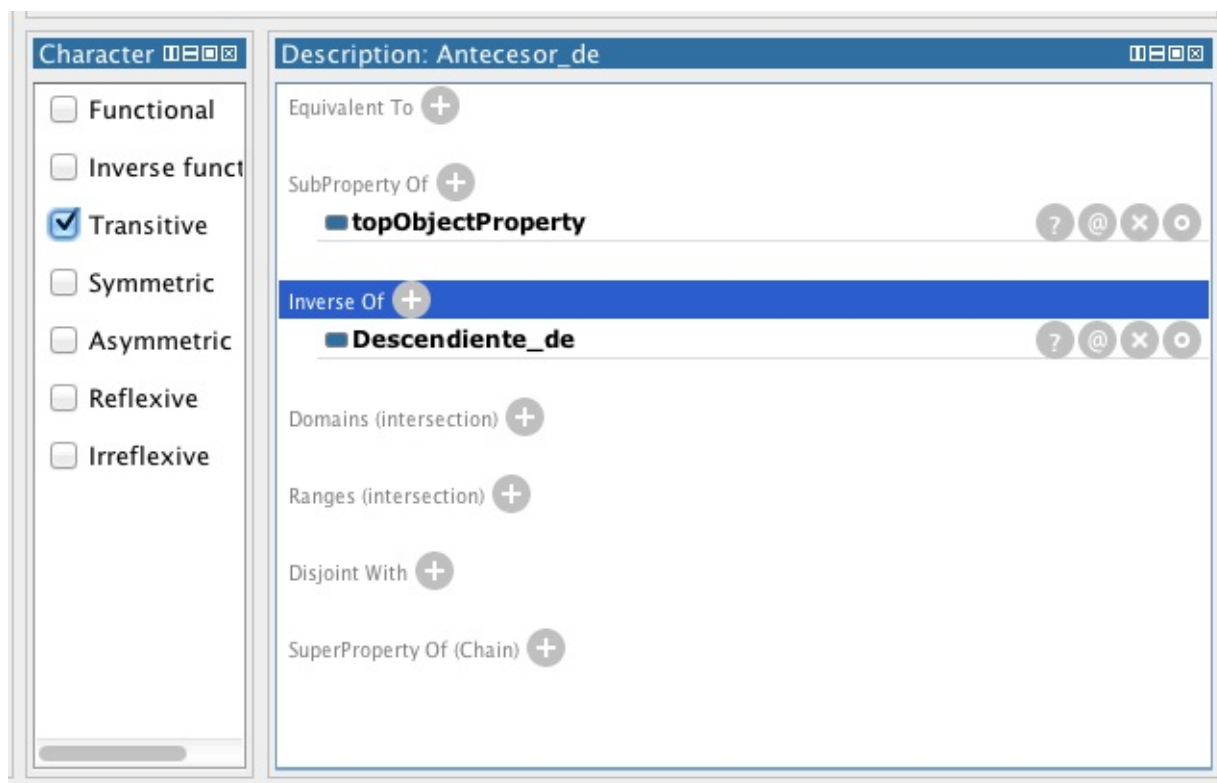
Name:

IRI:

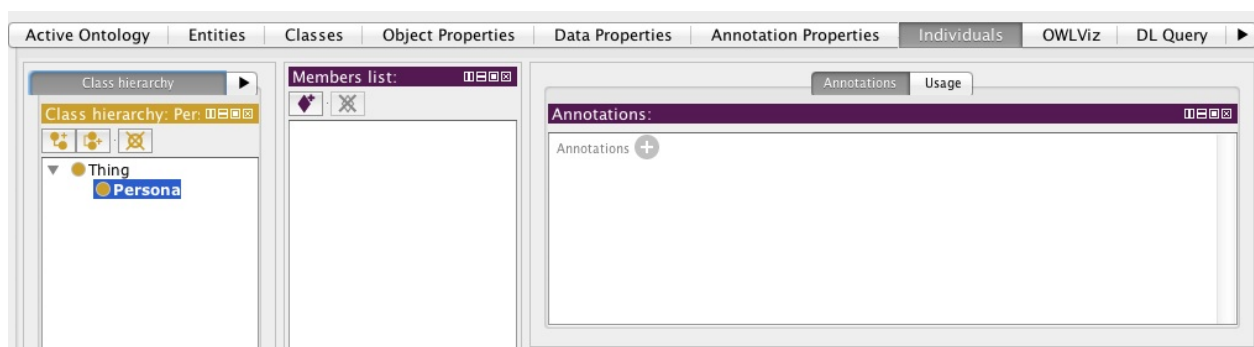
Volvemos a hacer lo mismo para crear la propiedad "Antecesor\_de".



Hacemos click en "Antecesor\_de" y marcamos "Transitive" en el cuadro "Character". Esto hará que el razonador reconozca la propiedad como transitiva. Es decir, si A es antecesor de B y B es antecesor de C, automáticamente pueda deducir que A es antecesor de C. En el cuadro de "Description" hacemos click en el "+" al lado de "Inverse Of" y señalamos "Descendiente\_de". Así le estamos diciendo que una propiedad es la inversa de la otra. En "Descendiente\_de" se marcará "Antecesor\_de" como propiedad inversa de forma automática.



Nos vamos ahora a la pestaña "Individuals" para crear las distintas instancias de Persona.



Hacemos click en el botón del cuadro "Members list" y creamos una persona que se llame "Juan".

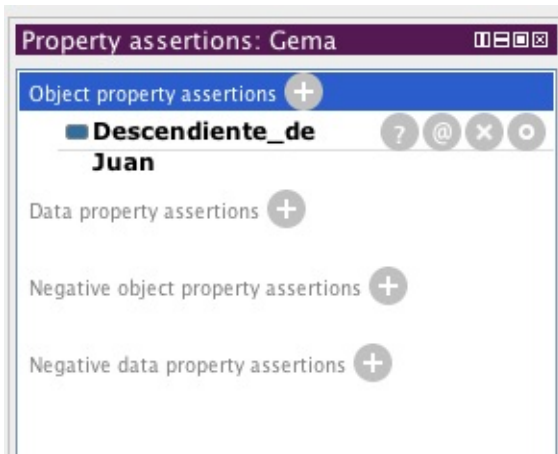
Name:

IRI:

Repetimos el procedimiento para crear a "Gema" y "Raúl".



Nuestra intención es indicar que Juan es el padre de Gema, y a su vez Gema es madre de Raúl. Para ello, seleccionamos a Gema y en el cuadro "Property assertions" pulsamos en el "+" al lado de "Object property assertions" y seleccionamos "Descendiente\_de" y "Juan".



Repetimos el mismo procedimiento con Raúl para hacerlo "Descendiente\_de" Gema.



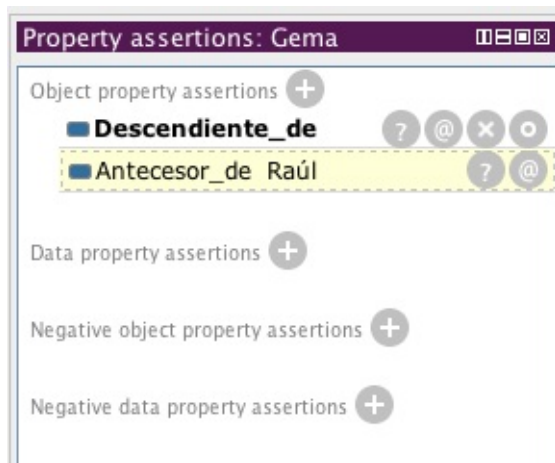
Ahora arrancamos el razonador. Para ello vamos al menú principal del programa, a "Reasoner". Por defecto estará señalado el razonador "HermiT". Pulsamos en "Start reasoner".



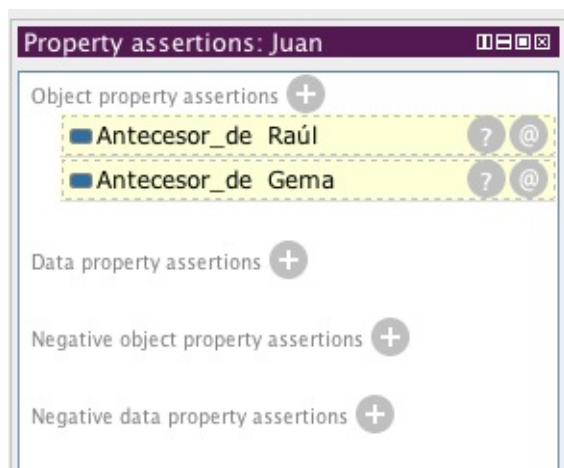
Si vemos las propiedades de Raúl haciendo click en su nombre, veremos que el razonador ha deducido que es "Descendiente\_de" Juan por la propiedad transitiva.



Si vemos las propiedades de Gema haciendo click en su nombre, veremos que el razonador ha deducido que es "Antecesor\_de" Raúl porque ser propiedad inversa.



Por último, hacemos click en el nombre de Juan para comprobar que el razonador ha deducido correctamente que es antecesor de los otros dos, por propiedad inversa y por transitividad.



Con esto termina nuestro ejemplo. Os animamos a seguir la guía indicada al inicio del capítulo que muestra un ejemplo mucho más completo del potencial de Protégé.

## Ejemplo de extracción de RDFa de una página

El conjunto de extensiones [RDFa](#) nos permite introducir semántica en un contenido web. De forma que la información sea entendible por seres humanos y a la vez incorpore semántica que pueda procesar una máquina. [GRDDL](#) es un método para extraer de una página web con contenido RDFa el archivo RDF asociado. Es decir, un fichero RDF con la información semántica de la página.

En este capítulo vamos a ver de forma práctica cómo se aplica GRDDL a una URL para obtener el archivo RDF asociado. Para ello, existen varias [implementaciones](#) en distintos lenguajes. Veremos dos métodos, uno con un servicio online y otro con un cliente escrito en Python.

## Extraer RDF usando un servicio web

La URL de la que vamos a extraer el archivo RDF es el ejemplo que está accesible [aquí](#). Utilizaremos el servicio web que puede usar el método GRDDL para extraer el RDF que está disponible [aquí](#).

### Redland Raptor RDF Parser Demonstration

by [Dave Beckett](#)

This is a demonstration of using [Raptor](#) to parse various syntaxes into RDF triples. This was written using the [Redland Perl](#) language binding.

Firstly choose a syntax then either enter a URI of some content or put it in the text box and run the parse.

Syntax: ☒ RDF/XML ☐ N-Triples ☐ Turtle ☐ RSS Tag Soup ☐ GRDDL ☐ Guess via MIME type and URI

URI of content to parse

or content

Run Parser

No data given.

Seleccionamos GRDDL y ponemos la URL que vamos a parsear. También podríamos pegar directamente el código fuente de la página en la ventana más grande.



## Redland Raptor RDF Parser Demonstration

by [Dave Beckett](#)

This is a demonstration of using [Raptor](#) to parse various syntaxes into RDF triples. This was written using the [Redland Perl](#) language binding.

Firstly choose a syntax then either enter a URI of some content or put it in the text box and run the parse.

Syntax: ☐ RDF/XML ☐ N-Triples ☐ Turtle ☐ RSS Tag Soup ☒ GRDDL ☐ Guess via MIME type and URI

URI of content to parse

or content

Por último, pulsamos en "Run Parser" y nos aparecerá una tabla con la información RDF diferenciando entre sujeto, predicado y objeto.

GRDDL Parsing Results

Subject	Predicate	Object
<a href="https://dl.dropboxusercontent.com/u/8262273/Ejemplos%20Linked%20Data/ejemplo_grddl.html#a1">https://dl.dropboxusercontent.com/u/8262273/Ejemplos%20Linked%20Data/ejemplo_grddl.html#a1</a>	<a href="http://xmlns.com/foaf/0.1/homepage">http://xmlns.com/foaf/0.1/homepage</a>	<a href="http://www.ihmc.us/users/user.php?UserID=42">http://www.ihmc.us/users/user.php?UserID=42</a>
<a href="https://dl.dropboxusercontent.com/u/8262273/Ejemplos%20Linked%20Data/ejemplo_grddl.html#a1">https://dl.dropboxusercontent.com/u/8262273/Ejemplos%20Linked%20Data/ejemplo_grddl.html#a1</a>	<a href="http://xmlns.com/foaf/0.1/name">http://xmlns.com/foaf/0.1/name</a>	Patrick Hayes
<a href="https://dl.dropboxusercontent.com/u/8262273/Ejemplos%20Linked%20Data/ejemplo_grddl.html#a1">https://dl.dropboxusercontent.com/u/8262273/Ejemplos%20Linked%20Data/ejemplo_grddl.html#a1</a>	<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#type">http://www.w3.org/1999/02/22-rdf-syntax-ns#type</a>	<a href="http://xmlns.com/foaf/0.1/Person">http://xmlns.com/foaf/0.1/Person</a>
<a href="http://www.w3.org/TR/2004/REC-rdf-mt-20040210/">http://www.w3.org/TR/2004/REC-rdf-mt-20040210/</a>	<a href="http://purl.org/dc/elements/1.1/creator">http://purl.org/dc/elements/1.1/creator</a>	<a href="https://dl.dropboxusercontent.com/u/8262273/Ejemplos%20Linked%20Data/ejemplo_grddl.html#a1">https://dl.dropboxusercontent.com/u/8262273/Ejemplos%20Linked%20Data/ejemplo_grddl.html#a1</a>
<a href="http://www.w3.org/TR/2004/REC-rdf-mt-20040210/">http://www.w3.org/TR/2004/REC-rdf-mt-20040210/</a>	<a href="http://purl.org/dc/elements/1.1/title">http://purl.org/dc/elements/1.1/title</a>	RDF Semantics - W3C Recommendation 10 February 2004

## Extraer RDF usando un cliente

Utilizaremos un cliente en bash, aunque podríamos utilizar cualquiera de las implementaciones de GRDDL. En este caso será [grddler.sh](#). Para poder utilizarlo, necesitaremos tener instalados los paquetes python, xsltproc y md5sum.

El uso es muy sencillo, sólo tendremos que ejecutar el script en bash pasándole como argumentos la URL de la que queremos extraer el rdf y el archivo donde queremos almacenar el resultado. Podría ser algo como lo siguiente:

```
bash ./grddler.sh https://davidr.gitbooks.io/ejemplos-de-linked-data/content/ejemplo_grddl.html > output.rdf
```

Con esto ya tendremos nuestro RDF extraído y damos por terminado el capítulo.

# Ejemplos de consultas y exploración en DBpedia

En este capítulo veremos como podemos realizar consultas SPARQL en DBpedia (utilizaremos la versión en español, pero la versión en inglés es similar) y cómo podríamos navegar entre los datos a través de las URI.

## Consultas SPARQL

Estas consultas de ejemplo las realizaremos desde el [endpoint](#) de la DBpedia en castellano.

### Listado de películas en español

En esta primera consulta obtendremos el listado de URIs de las películas en español recogidas en la DBpedia. Para ello, ejecutaremos la siguiente consulta:

```
PREFIX dcterms: <http://purl.org/dc/terms/>
SELECT ?Pelicula WHERE {
  ?Pelicula dcterms:subject <http://es.dbpedia.org/resource/Categoría:Películas_en_español> .
}
```

En las primeras líneas de cualquier consulta debemos indicar las ontologías de las propiedades que vamos a utilizar. En este caso, vamos a utilizar el vocabulario "dcterms:" de Dublin Core. Así que en la primera línea declaramos ese prefijo con su URL correspondiente.

La sentencia "SELECT" nos permite elegir el atributo o los atributos que van a mostrarse en el resultado. En este caso hemos señalado que queremos mostrar lo que guardemos en "?Pelicula". Podríamos usar un "\*" para que aparezcan todos los atributos que hemos usado en la consulta.

Dentro de las llaves de la estructura "WHERE" es donde debemos indicar las condiciones que nos devolverán un resultado. Cada condición debe expresarse con tres elementos (sujeto, predicado y objeto). En este caso, estamos seleccionando los elementos que hemos llamado "?Pelicula" cuya propiedad "dcterms:subject" apunte a la categoría de las películas en español (las URL se ponen entre <>). Es decir, seleccioname todos los elementos que a partir de ahora voy a llamar "?Pelicula" que tengan como valor en la propiedad "dcterms:subject" la categoría de películas en español.

Esta consulta nos devolverá un listado de las URIs de los elementos que cumplan esa condición, por lo que nos mostrará todas las películas incluidas en esa categoría. Algo parecido a lo que se muestra en la siguiente imagen.

Película
<a href="http://es.dbpedia.org/resource/Cleopatra_era_Cándida">http://es.dbpedia.org/resource/Cleopatra_era_Cándida</a>
<a href="http://es.dbpedia.org/resource/Fando_y_Lis">http://es.dbpedia.org/resource/Fando_y_Lis</a>
<a href="http://es.dbpedia.org/resource/Fiebre_de_amor_(película)">http://es.dbpedia.org/resource/Fiebre_de_amor_(película)</a>
<a href="http://es.dbpedia.org/resource/Plácido">http://es.dbpedia.org/resource/Plácido</a>
<a href="http://es.dbpedia.org/resource/Traffic_(película)">http://es.dbpedia.org/resource/Traffic_(película)</a>
<a href="http://es.dbpedia.org/resource/Amor_a_la_española">http://es.dbpedia.org/resource/Amor_a_la_española</a>
<a href="http://es.dbpedia.org/resource/Domingo_de_carnaval">http://es.dbpedia.org/resource/Domingo_de_carnaval</a>
<a href="http://es.dbpedia.org/resource/Faustino_Mayta_visita_a_su_prima">http://es.dbpedia.org/resource/Faustino_Mayta_visita_a_su_prima</a>
<a href="http://es.dbpedia.org/resource/La_cruel_Martina">http://es.dbpedia.org/resource/La_cruel_Martina</a>
<a href="http://es.dbpedia.org/resource/La_patota">http://es.dbpedia.org/resource/La_patota</a>

## Director de una película

Partimos del resultado anterior. Imagínate que ahora queremos saber cuál es el director de alguna de las películas, por ejemplo, Solas. Este dato se especifica en la propiedad "dbpedia-owl:director" de la película en concreto. Realizamos la siguiente consulta:

```
PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>
SELECT ?Nombre WHERE {
  <http://es.dbpedia.org/resource/Solas_(película)> dbpedia-owl:director ?Director .
  ?Director dbpedia-owl:abstract ?Nombre .
}
```

Podemos ver como primero volvemos a indicar la dirección de la ontología que vamos a usar. Luego seleccionamos mostrar el valor de "?Nombre" (Ahora veremos que estamos almacenando ahí). Esta vez tenemos dos condiciones en "WHERE", cuando hay más de una condición deben terminar con un espacio y un punto. En la primera le estamos diciendo que almacene en "?Director" el valor de la propiedad "dbpedia-owl:director" de la URI que se refiere a la película Solas (Devuelta en la consulta anterior). El problema es que ahora en "?Director" hay otra URI, la que apunta al director de la película en concreto. Para mostrar un resultado más legible y completo, en la segunda condición le vamos a pedir el abstract (resumen) de ese director. Para ello ponemos la URI de "?Director" en sujeto, que la propiedad que mire sea "dbpedia-owl:abstract" y el valor lo almacene en "?Nombre". Y esto último es lo que elegimos mostrar con SELECT. El resultado será el siguiente:

Nombre
"Benito Zambrano (n. Lebrija, 20 de marzo de 1965) es un guionista y director de cine español."@es

## Películas españolas estrenadas después del año 2000

```
PREFIX dcterms: <http://purl.org/dc/terms/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?Nombres ?Fecha WHERE {
  ?Pelicula dcterms:subject <http://es.dbpedia.org/resource/Categoría:Películas_en_español> .
  ?Pelicula foaf:name ?Nombres .
  ?Pelicula <http://es.dbpedia.org/property/estreno> ?Fecha
  FILTER (?Fecha > 2000)
}
```

En esta ocasión utilizaremos dos ontologías diferentes, "dcterms:" y "foaf:", por lo que lo indicamos en las primeras líneas. Esta vez tenemos tres sentencias en "WHERE". La primera ya la conocemos, es exactamente igual a la primera consulta realizada en el capítulo. Estamos poniendo en "?Pelicula" todas las URIs de las películas de la categoría "Películas en español". En la segunda, podemos ver como almacenamos en "?Nombres" los valores a los que apunta la propiedad "foaf:name" en cada película. Esta propiedad almacena el nombre de la película en un formato de texto plano. Por último, guardamos en "?Fecha" el contenido de la propiedad "<http://es.dbpedia.org/property/estreno>". Esta propiedad indica la fecha de estreno de la película. Por lo tanto, ya tenemos los nombres con sus correspondientes fechas de estreno. Pero aún vamos un paso más allá, y ponemos un filtro (FILTER). Esto nos permite restringir los resultados a ciertas condiciones. En este caso, indicamos que la fecha debe ser mayor que 2000, para que se muestren únicamente las películas estrenadas después de este año. En "SELECT" seleccionamos "?Nombre" y "?Fecha" por lo que el resultado será algo parecido a la siguiente imagen.

Nombres	Fecha
"A Better Life"@es	2011
"El Fin"@es	2012
"Juan de los Muertos"@es	2010
"XP3D"@es	2011
"Azul y no tan rosa"@es	2012
"Fin"@es	2012
"Libre te quiero"@es	2012
"Mágica"@es	2012
"Magic (Norteamérica)"@es	2012
"No me ama"@es	2010
"She doesn't love me (Norteamérica)"@es	2010
"El cielo dividido"@es	2006
"El cielo gira"@es	2004
"Hasta el viento tiene miedo"@es	2007
"La última cima"@es	2010
"Mientras duermes"@es	2011
"Zuquillo Expres"@es	2010

A continuación seguiremos refinando la misma búsqueda. Podrás comprobar en la imagen que los datos no están ordenados de ninguna forma. En la siguiente consulta ordenaremos los resultados por fecha de estreno:

```
PREFIX dcterms: <http://purl.org/dc/terms/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?Nombres ?Fecha WHERE {
  ?Pelicula dcterms:subject <http://es.dbpedia.org/resource/Categoría:Películas_en_español> .
  ?Pelicula foaf:name ?Nombres .
  ?Pelicula <http://es.dbpedia.org/property/estreno> ?Fecha
  FILTER (?Fecha > 2000)
} ORDER BY (?Fecha)
```

Como puedes ver, fuera de la estructura de "WHERE" podemos poner una cláusula "ORDER BY" para indicar si queremos ordenar los resultados por una o varias variables, en orden ascendente o descendente. Tendremos ahora una respuesta parecida a la siguiente imagen.

"El asesino de la escuela"@es	2001
"School Killer"@es	2001
"Autonomía"@es	2001
"Asesinato en El Meneo"@es	2001
"En la cuerda floja"@es	2001
"Ten Minutes Older"@es	2002
"Embrujo de Shanghai"@es	2002
"Chile, los héroes están fatigados"@es	2002
"Paloma de papel"@es	2003
"Lo mejor que le puede pasar a un cruasán"@es	2003
"La flaqueza del bolchevique"@es	2003
"No hay tierra sin dueño"@es	2003
"Cesante"@es	2003
"El cielo gira"@es	2004

Por último, podemos limitar también la cantidad de resultados a mostrar. En la siguiente consulta estamos limitando a 20 resultados:

```
PREFIX dcterms: <http://purl.org/dc/terms/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?Nombres ?Fecha WHERE {
  ?Pelicula dcterms:subject <http://es.dbpedia.org/resource/Categoría:Películas_en_español> .
  ?Pelicula foaf:name ?Nombres .
  ?Pelicula <http://es.dbpedia.org/property/estreno> ?Fecha
  FILTER (?Fecha > 2000)
} ORDER BY (?Fecha)
LIMIT 20
```

Para ello sólo es necesario añadir "LIMIT" y el número de resultados que queremos mostrar.

## Consultar marido/mujer

En la siguiente consulta estamos preguntando por el esposo de Concha Velasco. Igual podríamos preguntar por los hijos, padres, parejas de los hijos o cualquier parentesco familiar. Siempre que esté recogido en la Wikipedia en el formato adecuado, podremos consultarlo.

```
PREFIX esdbp: <http://es.dbpedia.org/property/>
PREFIX esres: <http://es.dbpedia.org/resource/>
SELECT ?esposo WHERE {
  esres:Concha_Velasco    esdbp:cónyuge    ?esposo .
}
```



En este caso el resultado es una URI a la página en DBpedia del marido de Concha Velasco.

<b>esposo</b>
<a href="http://es.dbpedia.org/resource/Paco_Marsó">http://es.dbpedia.org/resource/Paco Marsó</a>

## Grupos musicales en un rango de años

En esta consulta pediremos los grupos musicales de rock que estuvieron en activo en los años 70. Definimos como tipo "MusicalArtist" según el vocabulario de la DBpedia y que el estilo sea "Rock". Hay que tener en cuenta que sólo aparecerán los que estuvieron en activo durante esos 10 años. Si se mantuvieron en activo hasta más tarde, no aparecerán.

```

PREFIX esdbpp: <http://es.dbpedia.org/property/>
PREFIX esdbpr: <http://es.dbpedia.org/resource/>
SELECT ?grupo WHERE{
    ?grupo rdf:type dbpedia-owl:MusicalArtist .
    ?grupo dbpedia-owl:activeYearsStartYear ?inicio .
    ?grupo dbpedia-owl:activeYearsEndYear ?fin .
    ?grupo esdbpp:estilo esdbpr:Rock .
    FILTER ( (?inicio > "1970-01-01T00:00:00Z"^^xsd:dateTime && ?inicio < "1980-01-01T00:00:00Z"^^xsd:dateTime ) ||
            (?fin > "1970-01-01T00:00:00Z"^^xsd:dateTime && ?fin < "1980-01-01T00:00:00Z"^^xsd:dateTime ) ||
            (?inicio < "1970-01-01T00:00:00Z"^^xsd:dateTime && ?fin > "1980-01-01T00:00:00Z"^^xsd:dateTime )
          )
    }ORDER BY DESC(?inicio)
LIMIT 20

```

Hemos ordenado de forma descendente por el año de inicio y también hemos puesto un límite de 20 resultados para mostrarlos en la siguiente imagen.

grupo
<a href="http://es.dbpedia.org/resource/Steve_Conte">http://es.dbpedia.org/resource/Steve Conte</a>
<a href="http://es.dbpedia.org/resource/Antonio_Vega">http://es.dbpedia.org/resource/Antonio Vega</a>
<a href="http://es.dbpedia.org/resource/R.E.M.">http://es.dbpedia.org/resource/R.E.M.</a>
<a href="http://es.dbpedia.org/resource/Radio_Futura">http://es.dbpedia.org/resource/Radio Futura</a>
<a href="http://es.dbpedia.org/resource/PVP_(banda)">http://es.dbpedia.org/resource/PVP (banda)</a>
<a href="http://es.dbpedia.org/resource/Spinetta_Jade">http://es.dbpedia.org/resource/Spinetta Jade</a>
<a href="http://es.dbpedia.org/resource/Tito_Fargo">http://es.dbpedia.org/resource/Tito Fargo</a>
<a href="http://es.dbpedia.org/resource/Sabino_Méndez">http://es.dbpedia.org/resource/Sabino Méndez</a>
<a href="http://es.dbpedia.org/resource/Thomas_Anders">http://es.dbpedia.org/resource/Thomas Anders</a>
<a href="http://es.dbpedia.org/resource/Bow_Wow_Wow">http://es.dbpedia.org/resource/Bow Wow Wow</a>
<a href="http://es.dbpedia.org/resource/Antonio_Flores_(músico)">http://es.dbpedia.org/resource/Antonio Flores (músico)</a>
<a href="http://es.dbpedia.org/resource/Los_Rápidos">http://es.dbpedia.org/resource/Los Rápidos</a>
<a href="http://es.dbpedia.org/resource/Geniol_con_Coca">http://es.dbpedia.org/resource/Geniol con Coca</a>
<a href="http://es.dbpedia.org/resource/The_Revolution">http://es.dbpedia.org/resource/The Revolution</a>
<a href="http://es.dbpedia.org/resource/Kul_de_Mandril">http://es.dbpedia.org/resource/Kul de Mandril</a>
<a href="http://es.dbpedia.org/resource/Gary_Foote_(saxofonista)">http://es.dbpedia.org/resource/Gary Foote (saxofonista)</a>
<a href="http://es.dbpedia.org/resource/Ship_(banda)">http://es.dbpedia.org/resource/Ship (banda)</a>
<a href="http://es.dbpedia.org/resource/What_Is_This%3F">http://es.dbpedia.org/resource/What Is This%3F</a>
<a href="http://es.dbpedia.org/resource/The_Fastbacks">http://es.dbpedia.org/resource/The Fastbacks</a>
<a href="http://es.dbpedia.org/resource/Fischer-Z">http://es.dbpedia.org/resource/Fischer-Z</a>

## Exploración de los datos


Si entramos en cualquier URI de la DBpedia, veremos todos los enlaces que están relacionados con esta entidad. Por ejemplo, en la siguiente imagen podemos ver un fragmento de la URI de [Benito Zambrano](http://es.dbpedia.org/resource/Benito_Zambrano).

**About: Benito Zambrano**  
 An Entity of Type : [Concept](#), from Named Graph : <http://es.dbpedia.org>, within Data Space : [es.dbpedia.org](#)

Benito Zambrano (n. Lebrija, 20 de marzo de 1965) es un guionista y director de cine español.

Property	Value
<a href="#">dbpedia-owl:abstract</a>	<ul style="list-style-type: none"> <li>Benito Zambrano (n. Lebrija, 20 de marzo de 1965) es un guionista y director de cine español.</li> </ul>
<a href="#">dbpedia-owl:wikiPageID</a>	<ul style="list-style-type: none"> <li>426877 (xsd:integer)</li> </ul>
<a href="#">dbpedia-owl:wikiPageLength</a>	<ul style="list-style-type: none"> <li>2054 (xsd:integer)</li> </ul>
<a href="#">dbpedia-owl:wikiPageOutDegree</a>	<ul style="list-style-type: none"> <li>44 (xsd:integer)</li> </ul>
<a href="#">dbpedia-owl:wikiPageRevisionID</a>	<ul style="list-style-type: none"> <li>69353485 (xsd:integer)</li> </ul>
<a href="#">dbpedia-owl:wikiPageWikiLink</a>	<ul style="list-style-type: none"> <li><a href="#">dbpedia:Padre_coraje_(miniserie)</a></li> <li><a href="#">dbpedia:10_de_enero</a></li> <li><a href="#">dbpedia:1965</a></li> <li><a href="#">dbpedia:1987</a></li> <li><a href="#">dbpedia:1988</a></li> <li><a href="#">dbpedia:1989</a></li> <li><a href="#">dbpedia:1990</a></li> <li><a href="#">dbpedia:1991</a></li> <li><a href="#">dbpedia:1993</a></li> <li><a href="#">dbpedia:1995</a></li> <li><a href="#">dbpedia:1999</a></li> <li><a href="#">dbpedia:2002</a></li> <li><a href="#">dbpedia:2005</a></li> <li><a href="#">dbpedia:20_de_marzo</a></li> <li><a href="#">dbpedia:Categoría:Directores_de_cine_de_Andalucía</a></li> <li><a href="#">dbpedia:Categoría:Directores_de_cine_de_España</a></li> <li><a href="#">dbpedia:Categoría:Ganadores_del_Premio_Goya_a_la_mejor_dirección_novel</a></li> <li><a href="#">dbpedia:Categoría:Ganadores_del_Premio_Goya_al_mejor_guion_original</a></li> <li><a href="#">dbpedia:Categoría:Guionistas_de_cine_de_España</a></li> <li><a href="#">dbpedia:Cuba</a></li> <li><a href="#">dbpedia:Director_de_cine</a></li> <li><a href="#">dbpedia:España</a></li> <li><a href="#">dbpedia:Premios_Goya</a></li> <li><a href="#">dbpedia:Premios_Oscar</a></li> <li><a href="#">dbpedia:Sevilla</a></li> <li><a href="#">dbpedia:Academia_de_las_Artes_y_las_Ciencias_Cinematográficas_de_España</a></li> <li><a href="#">dbpedia:Club_Atlético_de_Madrid</a></li> <li><a href="#">dbpedia:Guionista</a></li> <li><a href="#">dbpedia:Lebrija_(Sevilla)</a></li> <li><a href="#">dbpedia:San_Antonio_de_Jos_Baños</a></li> <li><a href="#">dbpedia:2011</a></li> <li><a href="#">dbpedia:2012</a></li> <li><a href="#">dbpedia:Escuela_Internacional_de_Cine_y_Televisión</a></li> <li><a href="#">dbpedia:Premios_Oscar</a></li> <li><a href="#">dbpedia:Habana,_Blues</a></li> <li><a href="#">dbpedia:La_voz_doméstica</a></li> </ul>

Podemos ver dos columnas, en la izquierda aparece cada propiedad y en la derecha los posibles valores de esa propiedad (descripciones u otras URIs). Así, por ejemplo, podemos ver que está relacionado con la página de los premios Goya y podemos navegar hasta ella.

About: Premios Goya	
An Entity of Type : <a href="#">Award</a> , from Named Graph : <a href="http://es.dbpedia.org">http://es.dbpedia.org</a> , within Data Space : <a href="es.dbpedia.org">es.dbpedia.org</a>	
	
<p>Los Premios Goya o Premios Anuales de la Academia son los galardones otorgados de forma anual por la Academia de las Artes y las Ciencias Cinematográficas de España, con la finalidad de premiar a los mejores profesionales en cada una de las distintas especialidades del sector. El premio consiste en un busto de Francisco de Goya realizado en bronce por el escultor José Luis Fernández.</p>	
Property	Value
<a href="#">dbpedia-owl:abstract</a>	<ul style="list-style-type: none"> <li>Los Premios Goya o Premios Anuales de la Academia son los galardones otorgados de forma anual por la Academia de las Artes y las Ciencias Cinematográficas de España, con la finalidad de premiar a los mejores profesionales en cada una de las distintas especialidades del sector. El premio consiste en un busto de Francisco de Goya realizado en bronce por el escultor José Luis Fernández. La primera estatilla que se entregó fue obra del escultor Miguel Ortiz Berrocal; se trataba de una escultura desmontable, saliendo del busto de Goya una cámara. Desde la 4ª edición, la estatilla fue encargada a José Luis Fernández, desarrollando el mismo busto de forma y peso más reducido. La ceremonia de entrega de los premios tiene lugar entre los últimos días de enero y primeros de febrero y sigue el formato de los Premios Óscar.</li> </ul>
<a href="#">dbpedia-owl:country</a>	<ul style="list-style-type: none"> <li><a href="#">dbpedia:España</a></li> </ul>
<a href="#">dbpedia-owl:thumbnail</a>	<ul style="list-style-type: none"> <li><a href="http://commons.wikimedia.org/wiki/Special:FilePath/Premio_Goya.jpg?width=300">http://commons.wikimedia.org/wiki/Special:FilePath/Premio_Goya.jpg?width=300</a></li> </ul>
<a href="#">dbpedia-owl:wikiPageExternalLink</a>	<ul style="list-style-type: none"> <li><a href="http://premiosgoya.academiacine.com/home/index.php">http://premiosgoya.academiacine.com/home/index.php</a></li> <li><a href="http://www.academiacine.com/">http://www.academiacine.com/</a></li> <li><a href="http://www.elmundo.es/cultura/2014/02/09/52f73c9e2704e04118b456e.html">http://www.elmundo.es/cultura/2014/02/09/52f73c9e2704e04118b456e.html</a></li> <li><a href="http://especiales.abc.es/premios-goya/">http://especiales.abc.es/premios-goya/</a></li> <li><a href="http://vimeo.com/36705510/">http://vimeo.com/36705510/</a></li> <li><a href="http://www.cadenasur.com/premios-goya/">http://www.cadenasur.com/premios-goya/</a></li> <li><a href="http://www.elmundo.es/especiales/premios_goya/">http://www.elmundo.es/especiales/premios_goya/</a></li> <li><a href="http://www.epaia.com/especial/premios-goya">http://www.epaia.com/especial/premios-goya</a></li> <li><a href="http://www.toycinema.com/especial/goya/">http://www.toycinema.com/especial/goya/</a></li> <li><a href="http://www.rve.es/noticias/los-goya/">http://www.rve.es/noticias/los-goya/</a></li> <li><a href="http://www.premiosgoya.com">http://www.premiosgoya.com</a></li> <li><a href="http://www.foocast.com">http://www.foocast.com</a></li> </ul>
<a href="#">dbpedia-owl:wikiPageID</a>	11000 (xsd:integer)
<a href="#">dbpedia-owl:wikiPageLength</a>	49096 (xsd:integer)
<a href="#">dbpedia-owl:wikiPageOutDegree</a>	677 (xsd:integer)
<a href="#">dbpedia-owl:wikiPageRevisionID</a>	74154701 (xsd:integer)
<a href="#">dbpedia-owl:wikiPageWikiLink</a>	<ul style="list-style-type: none"> <li><a href="#">dbpedia:Anexo:Premio_Goya_a_la_mejor_dirección</a></li> <li><a href="#">dbpedia:Alfonso_Pino</a></li> <li><a href="#">dbpedia:Anexo:Premio_Goya_a_la_mejor_fotografía</a></li> <li><a href="#">dbpedia:Anexo:Premio_Goya_al_mejor_diseño_de_vestuario</a></li> <li><a href="#">dbpedia:Anexo:Premio_Goya_al_mejor_montaje</a></li> <li><a href="#">dbpedia:Anexo:XXII edición de los Premios Goya</a></li> <li><a href="#">dbpedia:Academia de las artes y las ciencias del cine de Francia</a></li> <li><a href="#">dbpedia:Carmen Machi</a></li> <li><a href="#">dbpedia:El laberinto del fauno</a></li> <li><a href="#">dbpedia:José_Corbacho</a></li> <li><a href="#">dbpedia:Lolita_(cantante)</a></li> <li><a href="#">dbpedia:Más_allá_del_jardín</a></li> </ul>

Incluso podemos saber más sobre una propiedad. Por ejemplo, si pulsamos en "country", nos llevará a una página que nos dice que es la misma propiedad que "country" en la versión en inglés. Y en esa página podremos ver datos de la propiedad.

About: país	
An Entity of Type : <a href="#">Property</a> , from Named Graph : <a href="http://dbpedia.org/resource/classes#">http://dbpedia.org/resource/classes#</a> , within Data Space : <a href="dbpedia.org">dbpedia.org</a>	
The country where the thing is located.	
Property	Value
<a href="#">rdf:type</a>	<ul style="list-style-type: none"> <li><a href="#">rdf:Property</a></li> <li><a href="#">owl:ObjectProperty</a></li> </ul>
<a href="#">rdfs:comment</a>	<ul style="list-style-type: none"> <li>The country where the thing is located.</li> </ul>
<a href="#">rdfs:isDefinedBy</a>	<ul style="list-style-type: none"> <li><a href="http://dbpedia.org/ontology/">http://dbpedia.org/ontology/</a></li> </ul>
<a href="#">rdfs:label</a>	<ul style="list-style-type: none"> <li><a href="#">country</a></li> <li><a href="#">país</a></li> </ul>
<a href="#">rdfs:range</a>	<ul style="list-style-type: none"> <li><a href="#">dbpedia-owl:Country</a></li> </ul>
<a href="#">rdfs:subPropertyOf</a>	<ul style="list-style-type: none"> <li><a href="#">dul:hasLocation</a></li> </ul>
<a href="#">owl:equivalentProperty</a>	<ul style="list-style-type: none"> <li><a href="http://wikidata.dbpedia.org/resource/P17">http://wikidata.dbpedia.org/resource/P17</a></li> </ul>
<a href="#">wdrs:describedby</a>	<ul style="list-style-type: none"> <li><a href="#">dbpedia-owl:data/definitions.xml</a></li> <li><a href="#">dbpedia-owl:data/definitions.ttl</a></li> <li><a href="#">dbpedia-owl:data/definitions.jsonld</a></li> </ul>
<a href="#">prov:wasDerivedFrom</a>	<ul style="list-style-type: none"> <li><a href="http://mappings.dbpedia.org/index.php/OntologyProperty:country">http://mappings.dbpedia.org/index.php/OntologyProperty:country</a></li> </ul>
<a href="#">is http://open.vocab.org/terms/defines of</a>	<ul style="list-style-type: none"> <li><a href="http://dbpedia.org/ontology/">http://dbpedia.org/ontology/</a></li> </ul>
<a href="#">is http://open.vocab.org/terms/describes of</a>	<ul style="list-style-type: none"> <li><a href="#">dbpedia-owl:data/definitions.xml</a></li> <li><a href="#">dbpedia-owl:data/definitions.ttl</a></li> <li><a href="#">dbpedia-owl:data/definitions.jsonld</a></li> </ul>

Así, podríamos explorar y navegar por las distintas entidades de forma infinita, ya que son cíclicas. Puede ser útil para descubrir relaciones que no sabíamos que existían. Y para ver datos sobre cualquier entidad de una forma mucho más lógica y ordenada que leyendo una larga descripción en la Wikipedia.